

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

IN RE APPLICATION OF: Hideki AKASHIKA, et al.

GAU:

SERIAL NO: New Application

EXAMINER:

FILED: Herewith

FOR: CONTROL SYSTEM AND CONTROL METHOD, METHOD AND APPARATUS FOR PROCESSING INFORMATION, INFORMATION PROCESSING TERMINAL AND METHOD THEREOF, STORAGE MEDIUM, AND PROGRAM

REQUEST FOR PRIORITY

COMMISSIONER FOR PATENTS
ALEXANDRIA, VIRGINIA 22313

SIR:

- ☐ Full benefit of the filing date of U.S. Application Serial Number , filed , is claimed pursuant to the provisions of 35 U.S.C. §120.
- ☐ Full benefit of the filing date(s) of U.S. Provisional Application(s) is claimed pursuant to the provisions of 35 U.S.C. §119(e): Application No. Date Filed

☒ Applicants claim any right to priority from any earlier filed applications to which they may be entitled pursuant to the provisions of 35 U.S.C. §119, as noted below.

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

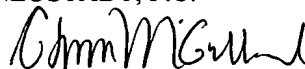
<u>COUNTRY</u>	<u>APPLICATION NUMBER</u>	<u>MONTH/DAY/YEAR</u>
Japan	2002-349994	December 2, 2002

Certified copies of the corresponding Convention Application(s)

- ☒ are submitted herewith
- ☐ will be submitted prior to payment of the Final Fee
- ☐ were filed in prior application Serial No. filed
- ☐ were submitted to the International Bureau in PCT Application Number
Receipt of the certified copies by the International Bureau in a timely manner under PCT Rule 17.1(a) has been acknowledged as evidenced by the attached PCT/IB/304.
- ☐ (A) Application Serial No.(s) were filed in prior application Serial No. filed ; and
- ☐ (B) Application Serial No.(s)
- ☐ are submitted herewith
- ☐ will be submitted prior to payment of the Final Fee

Respectfully Submitted,

OBLON, SPIVAK, McCLELLAND,
MAIER & NEUSTADT, P.C.



Bradley D. Lytle

Registration No. 40,073

Customer Number

22850

Tel. (703) 413-3000
Fax. (703) 413-2220
(OSMMN 05/03)

C. Irvin McClelland
Registration Number 21,124

503P1389 US00

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 2 年 1 2 月 2 日
Date of Application:

出 願 番 号 特 願 2 0 0 2 - 3 4 9 9 9 4
Application Number:
[ST. 10/C]: [J P 2 0 0 2 - 3 4 9 9 9 4]

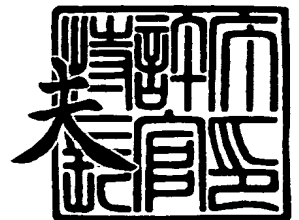
出 願 人 ソニー株式会社
Applicant(s):



2 0 0 3 年 8 月 2 6 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康



出証番号 出証特 2 0 0 3 - 3 0 6 9 5 8 6

【書類名】 特許願

【整理番号】 0290710210

【提出日】 平成14年12月 2日

【あて先】 特許庁長官殿

【国際特許分類】 H04L 29/06

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 赤鹿 秀樹

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 大嶋 拓哉

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 鈴木 唯史

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 三浦 敦史

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 太田 豊一

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 平野 義昭

【特許出願人】**【識別番号】** 000002185**【氏名又は名称】** ソニー株式会社**【代理人】****【識別番号】** 100082131**【弁理士】****【氏名又は名称】** 稲本 義雄**【電話番号】** 03-3369-6479**【手数料の表示】****【予納台帳番号】** 032089**【納付金額】** 21,000円**【提出物件の目録】****【物件名】** 明細書 1**【物件名】** 図面 1**【物件名】** 要約書 1**【包括委任状番号】** 9708842**【プルーフの要否】** 要

【書類名】 明細書

【発明の名称】 制御システムおよび方法、情報処理装置および方法、情報処理端末および方法、記録媒体、並びにプログラム

【特許請求の範囲】

【請求項 1】 ネットワークを介して接続される情報処理端末と情報処理装置からなる制御システムにおいて、

前記情報処理端末は、

機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求手段と、

前記要求手段による要求に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含むパケットを受信する受信手段と、

前記受信手段により受信された前記パケットに含まれる前記命令に基づいて、前記機器を制御する制御手段と

を備え、

前記情報処理装置は、

前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立手段と、

前記確立手段により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記パケットを前記情報処理端末に送信する送信手段と

を備えることを特徴とする制御システム。

【請求項 2】 ネットワークを介して接続される情報処理端末と情報処理装置からなる制御システムの制御方法において、

前記情報処理端末の情報処理方法は、

機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求ステップと、

前記要求ステップの処理に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含むパケットを受信する受信ステップと、

前記受信ステップの処理により受信された前記パケットに含まれる前記命令に基づいて、前記機器を制御する制御ステップと

を含み、

前記情報処理装置の情報処理方法は、

前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立ステップと、

前記確立ステップの処理により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記パケットを前記情報処理端末に送信する送信ステップと

を含むことを特徴とする制御方法。

【請求項 3】 ネットワークに接続される情報処理端末を介して、所定のタイミングで前記情報処理端末に接続される機器を制御する情報処理装置において、

前記情報処理端末に前記機器が接続された状態で行われる前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立手段と、

前記確立手段により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記機器の制御に関する命令を含む第 1 のパケットを前記情報処理端末に送信する送信手段と

を備えることを特徴とする情報処理装置。

【請求項 4】 前記送信手段により送信された前記命令に対応する、前記情報処理端末からの返信を含む第 2 のパケットを受信する受信手段をさらに備えることを特徴とする請求項 3 に記載の情報処理装置。

【請求項 5】 前記命令には、その命令に基づく処理が前記情報処理端末において失敗した場合に、次の命令に基づく処理を続行するか否かを表す情報が含まれる

ことを特徴とする請求項 3 に記載の情報処理装置。

【請求項 6】 前記送信手段は、同一のプロトコルに属する複数の前記命令を 1 つの前記第 1 のパケットに含めて送信する

ことを特徴とする請求項 3 に情報処理装置。

【請求項 7】 前記送信手段は、前記機器による処理の結果によって、前記機器に次に実行させる処理を前記情報処理端末に判断させる簡易プログラムを、前記第 1 のパケットと共に送信する

ことを特徴とする請求項 3 に記載の情報処理装置。

【請求項 8】 前記送信手段は、前記命令に基づく処理の対象となる前記機器を表す識別情報を含む前記第 1 のパケットを送信する

ことを特徴とする請求項 3 に記載の情報処理装置。

【請求項 9】 前記確立手段により確立された通信経路上に前記情報処理端末のファイアウォールが存在する場合、前記送信手段は、通信プロトコルとして HTTP を用い、前記第 1 のパケットを送信する

ことを特徴とする請求項 3 に記載の情報処理装置。

【請求項 10】 前記送信手段は、複数の前記第 1 のパケットを送信する期間、前記確立手段により確立された HTTP による通信リンクを継続させる

ことを特徴とする請求項 9 に記載の情報処理装置。

【請求項 11】 前記確立手段により確立された通信において、前記情報処理端末から所定の情報の送信が開始されるように、前記情報処理端末に指示する指示手段をさらに備える

ことを特徴とする請求項 3 に記載の情報処理装置。

【請求項 12】 ネットワークに接続される情報処理端末を介して、所定のタイミングで前記情報処理端末に接続される機器を制御する情報処理装置の情報処理方法において、

前記情報処理端末に前記機器が接続された状態で行われる前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立ステップと、

前記確立ステップの処理により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記機器の制御に関する命令を含むパケットを前記情報処理端末に送信する送信ステップと

を含むことを特徴とする情報処理方法。

【請求項 13】 ネットワークに接続される情報処理端末を介して、所定のタイミングで前記情報処理端末に接続される機器を制御する処理をコンピュータに行わせるプログラムの記録媒体において、

前記情報処理端末に前記機器が接続された状態で行われる前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立ステップと、

前記確立ステップの処理により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記機器の制御に関する命令を含むパケットを前記情報処理端末に送信する送信ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項 14】 ネットワークに接続される情報処理端末を介して、所定のタイミングで前記情報処理端末に接続される機器を制御する処理をコンピュータに行わせるプログラムにおいて、

前記情報処理端末に前記機器が接続された状態で行われる前記情報処理端末からの要求に応じて、前記情報処理端末との間で前記ネットワークを介した通信を確立する確立ステップと、

前記確立ステップの処理により前記情報処理端末との間で通信が確立されたとき、自分自身から、前記機器の制御に関する命令を含むパケットを前記情報処理端末に送信する送信ステップと

を含むことを特徴とするプログラム。

【請求項 15】 機器を遠隔制御する情報処理装置と、ネットワークを介して接続される情報処理端末において、

前記機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求手段と、

前記要求手段による要求に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含む第 1 のパケットを受信する受信手段と、

前記受信手段により受信された前記第 1 のパケットに含まれる前記命令に基づ

いて、前記機器を制御する制御手段と

を備えることを特徴とする情報処理端末。

【請求項 16】 前記制御手段による前記機器の制御結果を表す返信を含む第 2 のパケットを前記情報処理装置に送信する送信手段をさらに備える

ことを特徴とする請求項 15 に記載の情報処理端末。

【請求項 17】 機器を遠隔制御する情報処理装置と、ネットワークを介して接続される情報処理端末の情報処理方法において、

前記機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求ステップと、

前記要求ステップによる要求に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含むパケットを受信する受信ステップと、

前記受信ステップの処理により受信された前記パケットに含まれる前記命令に基づいて、前記機器を制御する制御ステップと

を含むことを特徴とする情報処理方法。

【請求項 18】 機器を遠隔制御する情報処理装置との間で行われる、ネットワークを介した情報処理をコンピュータに行わせるプログラムの記録媒体において、

前記機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求ステップと、

前記要求ステップによる要求に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含むパケットを受信する受信ステップと、

前記受信ステップの処理により受信された前記パケットに含まれる前記命令に基づいて、前記機器を制御する制御ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項 19】 機器を遠隔制御する情報処理装置との間で行われる、ネットワークを介した情報処理をコンピュータに行わせるプログラムにおいて、

前記機器が接続された状態で、前記情報処理装置に対して、前記ネットワークを介した通信の確立を要求する要求ステップと、

前記要求ステップによる要求に応じて確立された通信により前記情報処理装置から送信される、前記機器の制御に関する命令を含むパケットを受信する受信ステップと、

前記受信ステップの処理により受信された前記パケットに含まれる前記命令に基づいて、前記機器を制御する制御ステップと

を含むことを特徴とするプログラム。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、制御システムおよび方法、情報処理装置および方法、情報処理端末および方法、記録媒体、並びにプログラムに関し、特に、ネットワークを介して接続される機器の制御を、より確実に、かつ効率的に行うことができるようにする制御システムおよび方法、情報処理装置および方法、情報処理端末および方法、記録媒体、並びにプログラムに関する。

【0 0 0 2】

【従来の技術】

近年、携帯電話機などの通信端末を利用して、遠隔地に設置されている機器を制御するシステムが各種提案されている。

【0 0 0 3】

そのようなシステムとして、例えば、特許文献 1 には、外出先（外部）から電話で送信される暗証番号によって、家庭内に設置されたエアコンディショナやビデオ録画装置などの家電製品を制御するシステムが開示されている。

【0 0 0 4】

また、特許文献 2 には、制御端末と被制御端末の間にサーバが設置されており、そのサーバを経由して、制御端末が被制御端末の動作を制御するシステムが開示されている。

【0 0 0 5】

ところで、これらの制御システムにおいては、遠隔制御される装置が何らかの通信手段により常時接続されており、遠隔にある端末から必ず接続できる環境にあることを前提としている（例えば、特許文献1のシステムにおいては、家電製品がリモートコントローラおよびアダプタを介して電話回線に常時接続されている）。

【0006】

このように、制御する側の第1の装置と、制御される側の第2の装置が常に通信を行うことができる環境にある場合、いずれの装置も、自分自身が主導となって通信を開始させることが可能である。例えば、第1の装置から第2の装置に対してリクエストを含む各種のデータを任意のタイミングで送信したり、反対に、第2の装置から第1の装置に対して各種のデータを送信したりすることが可能である。

【0007】

従って、ネットワークを介して通信を行うことにより、どのタイミングであっても、装置を直接制御することができる。

【0008】

【特許文献1】

特開2000-184081号公報

【特許文献2】

特開2002-41378号公報

【0009】

【発明が解決しようとする課題】

しかしながら、このように、ネットワークに常時接続されている環境にない、例えば、ICカード、ディスプレイ、或いは、メモリなどのデバイスを遠隔操作する場合、それらのデバイスを管理するパーソナルコンピュータなどのクライアントからサーバにリクエストが送信され、サーバから返信されるリクエストの結果に基づいて、デバイスが操作されるという仕組みが一般的であり、サーバが主導となり、デバイスを直接操作するということが困難であるという課題があった。

【0010】

すなわち、デバイスがネットワークに常時接続されていないため、サーバからクライアントに任意のタイミングでアクセスし、サーバが主導となり、クライアントにより管理されているデバイスを制御することができない。

【0011】

具体的には、ネットワークに接続されている装置（例えば、ICカードリーダーライタやRF-ID受信装置）に近接または載置されることにより、所定のタイミングでサーバに接続されるICカードや非接触RF-IDなどのデバイスに対して、サーバが主導となって、デバイスの属性を変更させたり、または、デバイス进行操作したりすることは非常に困難である。

【0012】

また、従来の制御システムにおいては、サーバクライアント間でコマンドやレスポンスを送受信する場合、一般的に、1コマンド毎に、または1レスポンス毎に送受信が行われるため、サーバクライアント間での送受信の回数が多くなり、それに伴い、通信コストが高くなるという課題があった。

【0013】

コマンドやレスポンスの送受信が行われる毎に、ヘッダやフッタなどの情報が付加されるため、それらの送受信回数が増えるにつれて、1回の通信データ量が増え、さらに、1回の送受信に要する通信時間が長くなる。この課題は、通信速度が遅く、通信コストが比較的高い、例えば、携帯電話機などの端末を遠隔操作する場合、さらに深刻なものとなる。

【0014】

また、従来の制御システムにおいては、例えば、サーバから送信されてきたコマンドを受信したクライアントでエラーが発生した場合、エラーの内容にかかわらず、その時点で常に処理が中断されてしまい、エラーが発生した処理の次の処理をクライアントに実行させるには、サーバクライアント間で再接続し、コマンドを再度送信する必要があった。

【0015】

本発明はこのような状況に鑑みてなされたものであり、ネットワークを介して接続される機器の制御を、より確実に、かつ効率的に行うことができるようにす

るものである。

【0016】

【課題を解決するための手段】

本発明の制御システムの情報処理端末は、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立を要求する要求手段と、要求手段による要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含むパケットを受信する受信手段と、受信手段により受信されたパケットに含まれる命令に基づいて、機器を制御する制御手段とを備え、情報処理装置は、情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信を確立する確立手段と、確立手段により情報処理端末との間で通信が確立されたとき、自分自身から、パケットを情報処理端末に送信する送信手段とを備えることを特徴とする。

【0017】

本発明の制御システムの制御方法は、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立を要求する要求ステップと、要求ステップの処理に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含むパケットを受信する受信ステップと、受信ステップの処理により受信されたパケットに含まれる命令に基づいて、機器を制御する制御ステップと、情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信を確立する確立ステップと、確立ステップの処理により情報処理端末との間で通信が確立されたとき、自分自身から、パケットを情報処理端末に送信する送信ステップとを含むことを特徴とする。

【0018】

本発明の情報処理装置は、情報処理端末に機器が接続された状態で行われる情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信を確立する確立手段と、確立手段により情報処理端末との間で通信が確立されたとき、自分自身から、機器の制御に関する命令を含む第1のパケットを情報処理端末に送信する送信手段とを備えることを特徴とする。

【0019】

送信手段により送信された命令に対応する、情報処理端末からの返信を含む第 2 のパケットを受信する受信手段をさらに備えるようにすることができる。

【0020】

命令には、その命令に基づく処理が情報処理端末において失敗した場合に、次の命令に基づく処理を続行するか否かを表す情報が含まれるようにすることができる。

【0021】

送信手段は、同一のプロトコルに属する複数の命令を 1 つの第 1 のパケットに含めて送信するようにすることができる。

【0022】

送信手段は、機器による処理の結果によって、機器に次に実行させる処理を情報処理端末に判断させる簡易プログラムを、第 1 のパケットと共に送信するようにすることができる。

【0023】

送信手段は、命令に基づく処理の対象となる機器を表す識別情報を含む第 1 のパケットを送信するようにすることができる。

【0024】

確立手段により確立された通信経路上に情報処理端末のファイアウォールが存在する場合、送信手段は、通信プロトコルとして HTTP を用い、第 1 のパケットを送信するようにすることができる。

【0025】

送信手段は、複数の第 1 のパケットを送信する期間、確立手段により確立された HTTP による通信リンクを継続させるようにすることができる。

【0026】

確立手段により確立された通信において、情報処理端末から所定の情報の送信が開始されるように、情報処理端末に指示する指示手段をさらに備えるようにすることができる。

【0027】

本発明の情報処理装置の情報処理方法は、情報処理端末に機器が接続された状

態で行われる情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信を確立する確立ステップと、確立ステップの処理により情報処理端末との間で通信が確立されたとき、自分自身から、機器の制御に関する命令を含むパケットを情報処理端末に送信する送信ステップとを含むことを特徴とする。

【0028】

本発明の記録媒体に記録されているプログラム、および、本発明の第1のプログラムは、情報処理端末に機器が接続された状態で行われる情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信を確立する確立ステップと、確立ステップの処理により情報処理端末との間で通信が確立されたとき、自分自身から、機器の制御に関する命令を含むパケットを情報処理端末に送信する送信ステップとを含むことを特徴とする。

【0029】

本発明の情報処理端末は、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立を要求する要求手段と、要求手段による要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含む第1のパケットを受信する受信手段と、受信手段により受信された第1のパケットに含まれる命令に基づいて、機器を制御する制御手段とを備えることを特徴とする。

【0030】

制御手段による機器の制御結果を表す返信を含む第2のパケットを情報処理装置に送信する送信手段をさらに備えるようにすることができる。

【0031】

本発明の情報処理端末の情報処理方法は、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立を要求する要求ステップと、要求ステップによる要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含むパケットを受信する受信ステップと、受信ステップの処理により受信されたパケットに含まれる命令に基づいて、機器を制御する制御ステップとを含むことを特徴とする。

【 0 0 3 2 】

本発明の記録媒体に記録されているプログラム、および、本発明の第 2 のプログラムは、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立を要求する要求ステップと、要求ステップによる要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含むパケットを受信する受信ステップと、受信ステップの処理により受信されたパケットに含まれる命令に基づいて、機器を制御する制御ステップとを含むことを特徴とする。

【 0 0 3 3 】

本発明の制御システムおよび方法においては、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立が要求され、要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含むパケットが受信され、受信されたパケットに含まれる命令に基づいて、機器が制御される。また、情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信が確立され、情報処理端末との間で通信が確立されたとき、パケットが情報処理端末に送信される。

【 0 0 3 4 】

本発明の情報処理装置および方法、記録媒体、並びにプログラムにおいては、情報処理端末に機器が接続された状態で行われる情報処理端末からの要求に応じて、情報処理端末との間でネットワークを介した通信が確立され、情報処理端末との間で通信が確立されたとき、機器の制御に関する命令を含む第 1 のパケットが情報処理端末に送信される。

【 0 0 3 5 】

本発明の情報処理端末および方法、記録媒体、並びにプログラムにおいては、機器が接続された状態で、情報処理装置に対して、ネットワークを介した通信の確立が要求され、要求に応じて確立された通信により情報処理装置から送信される、機器の制御に関する命令を含む第 1 のパケットが受信される。また、受信された第 1 のパケットに含まれる命令に基づいて、機器が制御される。

【 0 0 3 6 】

【発明の実施の形態】

図1は、本発明を適用した制御システムの構成例を示す図である。

【0037】

ネットワーク1には、ネットワーク1に接続されるクライアント（クライアントを介してネットワーク1に接続されるデバイス）を遠隔制御するサーバ2およびサーバ4が接続されている。サーバ2には、クライアントとの間で行われる通信において、サーバ2と協働してデータの暗号化処理および復号処理、並びに認証処理を実行する暗号・認証用サーバ3が接続されている。

【0038】

サーバ2またはサーバ4によるデバイス（機器）の制御は、クライアントを介して行われる。

【0039】

ネットワーク1には、クライアントとしての携帯電話機5およびパーソナルコンピュータ6が接続されている。後に詳述するように、クライアント（携帯電話機5またはパーソナルコンピュータ6）と、サーバ（サーバ2またはサーバ4）の間で行われる通信においては、サーバが主導となってクライアントを制御することができるプロトコルが用いられる。

【0040】

図1の例においては、サーバ2またはサーバ4によりクライアントを介して制御されるデバイスとして、リーダライタ11、ICカード13、スピーカ14、およびディスプレイ15が携帯電話機5に接続または内蔵されている。

【0041】

また、携帯電話機5には、ICカード13のように携帯電話機5に直接接続または内蔵されるのではなく、近接または載置されたときに、リーダライタ11との間で電磁誘導を利用した通信を行う外部ICカード12がリーダライタ11を介して接続されている。

【0042】

同様に、図1においては、サーバ2またはサーバ4によりクライアントを介して制御されるデバイスとして、ICカード21、スピーカ22、およびディスプレ

イ 23、リーダライタ 24 がパーソナルコンピュータ 6 に接続されている。リーダライタ 24 には、所定のタイミングで、外部ICカード 25 が近接または載置される。

【0043】

これらのデバイス（リーダライタ 11、外部ICカード 12、ICカード 13、スピーカ 14、ディスプレイ 15、ICカード 21、スピーカ 22、ディスプレイ 23、リーダライタ 24、および外部ICカード 25）は、ネットワーク 1 にクライアントを介して常時接続されているものではなく、所定のタイミングで接続されるものである。

【0044】

当然、図 1 に示されるシステム構成は適宜変更可能である。例えば、携帯電話機 5 やパーソナルコンピュータ 6 以外にも、クライアントとしての PDA(Personal Digital Assistants)や、デバイスとしてのメモリなどがネットワーク 1 を介してサーバ 2 やサーバ 4 に接続されるようにしてもよい。

【0045】

図 2 は、図 1 のサーバ 2 の構成例を示すブロック図である。

【0046】

CPU (Central Processing Unit) 31 は、ROM (Read Only Memory) 32 に記憶されているプログラム、または、記憶部 38 から RAM (Random Access Memory) 33 にロードされたプログラムに従って各種の処理を実行する。RAM 33 にはまた、CPU 31 が各種の処理を実行する上において必要なデータなどが適宜記憶される。

【0047】

例えば、CPU 31 により所定の制御プログラムが実行され、図 1 の制御システムにおいて用いられるプロトコル（以下、適宜、制御プロトコルと称する）の通信によりクライアントを制御するサーバが実現される。

【0048】

CPU 31、ROM 32、および RAM 33 は、バス 34 を介して相互に接続されている。このバス 34 にはまた、入出力インタフェース 35 も接続されている。

【0049】

入出力インタフェース 35 には、キーボード、マウスなどよりなる入力部 36、CRT(Cathode Ray Tube)、LCD(Liquid Crystal Display)などよりなるディスプレイ、並びにスピーカなどよりなる出力部 37、ハードディスクなどより構成される記憶部 38、モデム、ターミナルアダプタなどより構成される通信部 39 が接続されている。通信部 39 は、ネットワーク 1 を介しての通信処理を行う。

【0050】

入出力インタフェース 35 にはまた、必要に応じてドライブ 40 が接続され、磁気ディスク 41、光ディスク 42、光磁気ディスク 43、或いは半導体メモリ 44 などが適宜装着され、それから読み出されたコンピュータプログラムが、必要に応じて記憶部 38 にインストールされる。

【0051】

サーバ 4 の構成、並びに、クライアントである携帯電話機 5 およびパーソナルコンピュータ 6 の構成は、図 2 に示されるサーバ 2 の構成と基本的に同様であるため、その説明は省略する。なお、図 2 は、必要に応じて、携帯電話機 5 またはパーソナルコンピュータ 6 の構成としても引用される。

【0052】

ここで、サーバとクライアントの間で行われる通信において用いられるプロトコルについて説明する。以下、適宜、サーバとクライアントの間で行われる通信において用いられるプロトコルを制御プロトコルと称する。

【0053】

この制御プロトコルにより、クライアントにおいて実行されている Web アプリケーションの背後で、クライアントとサーバ間の通信が行われる。なお、携帯電話機 5 などでは、Web アプリケーションは必要なく、クライアントとの機器通信のみの場合がある。この場合、例えば、携帯電話機 5 のアプリケーションが、表示用データを全て保持しており、デバイスの制御と同様に、アプリケーションやディスプレイに対して制御を行うことにより、表示等を行う。

【0054】

制御プロトコルを使用して Web アプリケーションを開発することにより、例え

ば、第1に、ビジネスロジックを1つの関数／メソッドの中に記述することができ（HTTP(Hyper Text Transfer Protocol)コネクションが毎回切れることを意識する必要がない）、第2に、サーバからクライアントに対してリクエストを送信させることができ（HTTPの「リクエストの送信元は常にクライアント」という制限を受けない）、第3に、クライアント上で任意の処理を実行させることができる（Applet, ActiveXコントロールのような制限を受けない）。

【0055】

制御プロトコルは、Data Transfer Protocolと、その他のサブプロトコルで構成される。例えば、サブプロトコルとして、Handshake Protocol, Farewell Protocol, Error Protocol, Update Entity Protocol, Application Data Transfer Protocol, Operate Entity Protocolの6つが規定される。

【0056】

具体的には、Data Transfer Protocolは、サブプロトコル電文の送受信手順を規定するものであり、Handshake Protocolは、コネクションの確立手順を規定するものである。また、Farewell Protocolは、コネクションの開放手順を規定するものであり、Error Protocolは、エラーの通知手順を規定するものである。

【0057】

さらに、Update Entity Protocolは、サーバがクライアントの状態を変更する手順を規定するものであり、Application Data Transfer Protocolは、アプリケーション固有のデータの送受信手順を規定するものであり、Operate Entity Protocolは、サーバがクライアントを操作する手順を規定するものである。

【0058】

これらのプロトコルを含む制御プロトコルは、データ通信の核（粋）となる部分だけを定義するものであり、所定のデバイス进行操作するなど、アプリケーション固有の処理を実行するには、そのアプリケーション用のエクステンション（拡張モジュール）が必要である。すなわち、制御プロトコルによって、制御プロトコル本体とエクステンションが協調して動作するための手順が定義される。

【0059】

図3は、制御プロトコルに基づく通信を実現するサーバ（例えば、サーバ2）

の機能構成例を示すブロック図である。図3に示される各機能ブロックは、例えば、サーバ2のCPU31により、所定の制御プログラムが実行されて実現される。

【0060】

制御部51は、Handshake Protocolに属するメッセージ（命令）を管理するHandshakeメッセージ管理部52、Farewell Protocolに属するメッセージを管理するFarewellメッセージ管理部55、Error Protocolに属するメッセージを管理するErrorメッセージ管理部58、Application Data Transfer Protocolに属するメッセージを管理するApplication Data Transferメッセージ管理部61、Update Entity Protocolに属するメッセージを管理するUpdate Entityメッセージ管理部64、および、Operate Entity Protocolに属するメッセージを管理するOperate Entityメッセージ管理部67を含む、サーバ2の機能全体を制御する。

【0061】

なお、Handshake Protocolに属するメッセージ（以下、適宜、Handshakeメッセージと称する）については図7を参照して、Farewell Protocolに属するメッセージ（以下、適宜、Farewellメッセージと称する）については図9を参照して、Error Protocolに属するメッセージ（以下、適宜、Errorメッセージと称する）については図11を参照して、Update Entity Protocolに属するメッセージ（以下、適宜、Update Entityメッセージと称する）については図14を参照して、Application Data Transfer Protocolに属するメッセージ（以下、適宜、Application Data Transferメッセージと称する）については図16を参照して、Operate Entity Protocolに属するメッセージ（以下、適宜、Operate Entityメッセージと称する）については図18を参照して、それぞれ詳述する。

【0062】

Handshakeメッセージ管理部52は、Handshakeメッセージに関する処理を管理し、例えば、クライアントからパケットに含められて送信されてきたHandshakeメッセージがHandshakeメッセージ受信制御部54から供給されてきたとき、そのメッセージに応じたHandshakeメッセージを生成し、クライアントに対して送信すべく、生成したHandshakeメッセージをHandshakeメッセージ送信制御部53

に出力する。

【0063】

Handshakeメッセージ送信制御部53は、図2の通信部39において行われる、携帯電話機5などのクライアントに対するHandshakeメッセージの送信を制御し、Handshakeメッセージ受信制御部54は、図2の通信部39において行われる、クライアントから送信されてきたHandshakeメッセージの受信を制御する。

【0064】

Farewellメッセージ管理部55は、Farewellメッセージに関する処理を管理し、例えば、クライアントからパケットに含められて送信されてきたFarewellメッセージがFarewellメッセージ受信制御部57から供給されてきたとき、そのメッセージに応じたFarewellメッセージを生成し、クライアントに対して送信すべく、生成したFarewellメッセージをFarewellメッセージ送信制御部56に出力する。

【0065】

Farewellメッセージ送信制御部56は、図2の通信部39において行われる、クライアントに対するFarewellメッセージの送信を制御し、Farewellメッセージ受信制御部57は、図2の通信部39において行われる、クライアントから送信されてきたFarewellメッセージの受信を制御する。

【0066】

Errorメッセージ管理部58は、Errorメッセージに関する処理を管理し、例えば、クライアントから送信されてきたパケットにフォーマットのエラーが存在することが制御部51により検出されたとき、それに対応するErrorメッセージを生成し、クライアントに対して送信すべく、生成したErrorメッセージをErrorメッセージ送信制御部59に出力する。

【0067】

Errorメッセージ送信制御部59は、図2の通信部39において行われる、クライアントに対するErrorメッセージの送信を制御し、Errorメッセージ受信制御部60は、図2の通信部39において行われる、クライアントから送信されてきたErrorメッセージの受信を制御する。

【 0 0 6 8 】

Application Data Transferメッセージ管理部 6 1 は、Application Data Transferメッセージに関する処理を管理し、例えば、クライアントからパケットに含められて送信されてきたApplication Data TransferメッセージがApplication Data Transferメッセージ受信制御部 6 3 から供給されてきたとき、そのメッセージに応じたApplication Data Transferメッセージを生成し、クライアントに対して送信すべく、生成したApplication Data TransferメッセージをApplication Data Transferメッセージ送信制御部 6 2 に出力する。

【 0 0 6 9 】

Application Data Transferメッセージ送信制御部 6 2 は、図 2 の通信部 3 9 において行われる、クライアントに対するApplication Data Transferメッセージの送信を制御する。具体的には、Application Data Transferメッセージ送信制御部 6 2 は、Application Data Transferメッセージ管理部 6 1 から供給されてきた所定の数のメッセージにFINISHDメッセージを付加し、1つのパケットとしてクライアントに送信する。

【 0 0 7 0 】

Application Data Transferメッセージ受信制御部 6 3 は、図 2 の通信部 3 9 において行われる、クライアントから送信されてきたApplication Data Transferメッセージの受信を制御する。具体的には、Application Data Transferメッセージ受信制御部 6 3 は、クライアントから送信されてきたパケットの終端を、それに含まれるFINISHDメッセージにより検出し、パケットに含まれるApplication Data TransferメッセージをApplication Data Transferメッセージ管理部 6 1 に出力する。

【 0 0 7 1 】

Update Entityメッセージ管理部 6 4 は、Update Entityメッセージに関する処理を管理し、例えば、制御部 5 1 から、所定のUpdate Entityメッセージをクライアントに対して送信することが指示されたとき、指示に対応するUpdate Entityメッセージを生成し、クライアントに対して送信すべく、生成したUpdate EntityメッセージをUpdate Entityメッセージ送信制御部 6 5 に出力する。

【 0 0 7 2 】

Update Entityメッセージ送信制御部 6 5 は、図 2 の通信部 3 9 において行われる、クライアントに対するUpdate Entityメッセージの送信を制御する。具体的には、Update Entityメッセージ送信制御部 6 5 は、Update Entityメッセージ管理部 6 4 から供給されてきた所定の数のメッセージにFINISHDメッセージを付加し、1つのパケットとしてクライアントに送信する。

【 0 0 7 3 】

Update Entityメッセージ受信制御部 6 6 は、図 2 の通信部 3 9 において行われる、クライアントから送信されてきたUpdate Entityメッセージの受信を制御する。具体的には、Update Entityメッセージ受信制御部 6 6 は、クライアントから送信されてきたパケットの終端を、それに含まれるFINISHDメッセージにより検出し、パケットに含まれるUpdate EntityメッセージをUpdate Entityメッセージ管理部 6 4 に出力する。

【 0 0 7 4 】

Operate Entityメッセージ管理部 6 7 は、Operate Entityメッセージに関する処理を管理し、例えば、制御部 5 1 から、所定のOperate Entityメッセージをクライアントに対して送信することが指示されたとき、指示に対応するOperate Entityメッセージを生成し、クライアントに対して送信すべく、生成したOperate EntityメッセージをOperate Entityメッセージ送信制御部 6 8 に出力する。

【 0 0 7 5 】

Operate Entityメッセージ送信制御部 6 8 は、図 2 の通信部 3 9 において行われる、クライアントに対するOperate Entityメッセージの送信を制御する。具体的には、Operate Entityメッセージ送信制御部 6 8 は、Operate Entityメッセージ管理部 6 7 から供給されてきた所定の数のメッセージにFINISHDメッセージを付加し、1つのパケットとしてクライアントに送信する。

【 0 0 7 6 】

Operate Entityメッセージ受信制御部 6 9 は、図 2 の通信部 3 9 において行われる、クライアントから送信されてきたOperate Entityメッセージの受信を制御する。具体的には、Operate Entityメッセージ受信制御部 6 9 は、クライアント

から送信されてきたパケットの終端を、それに含まれるFINISHDメッセージにより検出し、パケットに含まれるOperate EntityメッセージをOperate Entityメッセージ管理部 6 7 に出力する。

【 0 0 7 7 】

図 4 は、制御プロトコルに基づく通信を実現するクライアント（例えば、携帯電話機 5）の機能構成例を示すブロック図である。図 4 に示される構成は、基本的には図 3 に示されるサーバ側の構成と同様であるため、上述したものと重複する説明については適宜省略する。

【 0 0 7 8 】

図 4 に示される各構成は、例えば、携帯電話機 5 の CPU 3 1 により、所定の制御プログラムが実行されて実現される。

【 0 0 7 9 】

制御部 8 1 は、Handshakeメッセージ管理部 8 2、Farewellメッセージ管理部 8 5、Errorメッセージ管理部 8 8、Application Data Transferメッセージ管理部 9 1、Update Entityメッセージ管理部 9 4、および、Operate Entityメッセージ管理部 9 7 を含む、携帯電話機 5（クライアント）の機能全体を制御する。

【 0 0 8 0 】

例えば、制御部 8 1 は、サーバから送信され、Update Entityメッセージ受信制御部 9 6 において受信された所定のUpdate EntityメッセージがUpdate Entityメッセージ管理部 9 4 を介して供給されてきたとき、デバイス制御部 1 0 0 を制御し、供給されてきたUpdate Entityメッセージに基づく処理をデバイスに実行させる。

【 0 0 8 1 】

また、制御部 8 1 は、クライアント自身に接続されているデバイスにより所定の処理を実行させたとき、デバイスによる実行結果（例えば、処理が成功したか否かを表す情報）をデバイス制御部 1 0 0 から取得し、それをUpdate Entityメッセージ管理部 9 4 に出力する。

【 0 0 8 2 】

Update Entityメッセージ管理部 9 4 においては、処理結果に基づくレスポンス

スメッセージ (Update Entityメッセージ) が生成され、Update Entityメッセージ送信制御部 95 を介して、クライアントに対して送信される。

【0083】

デバイス制御部 100 は、Application Data Transferメッセージ受信制御部 93 により取得された Application Data Transferメッセージが制御部 81 に供給され、それに基づく処理をデバイスに実行させることが制御部 81 から指示されたとき、また、Update Entityメッセージ受信制御部 96 により取得された Update Entityメッセージが制御部 81 に供給され、それに基づく処理をデバイスに実行させることが制御部 81 から指示されたとき、或いは、Operate Entityメッセージ受信制御部 99 により取得された Operate Entityメッセージが制御部 81 に供給され、それに基づく処理をデバイスに実行させることが制御部 81 から指示されたとき、それらの指示に基づいて、クライアント自身に接続されているデバイスの動作を制御する。

【0084】

また、デバイス制御部 100 は、その制御結果 (デバイスによる処理結果) を制御部 81 に出力する。この処理結果に基づいて、Application Data Transferメッセージ管理部 91、Update Entityメッセージ管理部 94、または、Operate Entityメッセージ管理部 97 においてサーバに対するレスポンスが生成される。

【0085】

次に、図 5 を参照して、制御プロトコルの状態遷移について説明する。なお、それぞれの状態において行われる通信の詳細については後述するものとする。

【0086】

ステップ S1 において、クライアント (例えば、携帯電話機 5) の CPU 31 によりクライアントプログラムが起動され、状態遷移が開始される。起動時には、例えば、以下の各情報が必要となる。

【0087】

- ・サーバ URL (Uniform Resource Locator)

クライアントを制御するサーバの URL (例: <http://www.oge.com/webapp/Serve>)

r)

【0088】

- ・ビュー表示WebアプリケーションURL

アプリケーションの前面でエンドユーザに対してビュー（画面）を提供するWebアプリケーションのURL（例：http://www.oge.com/webapp/ViewDispatcher）

【0089】

なお、携帯電話機5等において、Webアプリケーションが必要ない場合、このURLは必要ない。

【0090】

- ・クッキー（オプション）

セッションを識別するためのクッキー

【0091】

クライアントは、サーバURLにより指定されたサーバ（例えば、サーバ2）に対して、指定されたセッションIDを使ってHTTPリクエストを送信する。上述した例においては、「www.oge.com」の所定のポート番号に対して、「/webapp/Server」に電文をPOSTするHTTPリクエストが送信される。

【0092】

なお、起動時にクッキー情報がセットされた場合、クライアントは、HTTPリクエストのCookieヘッダに、指定されたクッキー情報を付加する。

【0093】

サーバからの返信電文はHTTPレスポンスのコンテンツ部に格納される。

【0094】

起動後、クライアントは、ステップS2において、Handshakeサブプロトコルを使ってCLIENT_HELLOメッセージをサーバに送信し、通信の確立を要求する。CLIENT_HELLOメッセージが送信されてから、コネクションが確立されるまでの状態がHandshakeステートとされる（図5のstatel）。

【0095】

クライアントは、Handshake Protocolを終了したとき、ステップS3において、サーバに対してFINISHEDメッセージを送信する。これによりコネクションが確

立され、制御の主導権がサーバに移る。

【 0 0 9 6 】

なお、CLIENT_HELLO_DONEをクライアントからサーバへ送信することにより、コネクションが確立される場合もある（図 2 5 参照）。

【 0 0 9 7 】

すなわち、サーバからクライアントに対して所定のコマンド（メッセージ）が送信され、そのメッセージに基づく処理結果などを表すレスポンスがクライアントからサーバに返信される。コネクションが確立し、主導権がサーバに移った状態がNeutralステートとされる（図 5 のstate2）。

【 0 0 9 8 】

Neutralステートの状態にあるステップ S 4 において、サーバは、自分自身が主導的に、アプリケーションのロジックに従って、クライアントの状態制御、またはアプリケーション固有データの送受信を行う。

【 0 0 9 9 】

このとき、クライアントの状態制御には、Update Entityサブプロトコルが用いられ、アプリケーション固有データの送受信にはApplication Data Transferサブプロトコルが用いられる。また、クライアント側のデバイスの操作にはOperate Entityサブプロトコルが用いられる。

【 0 1 0 0 】

Neutralステートに限り、これらの3つのサブプロトコルのパケットを、任意の順序で、複数同時に送受信することが可能とされる。例えば、図 2 0 乃至図 2 2 を参照して後述するように、1つのパケットに含めることができるサブプロトコル種別は1種類であり（1種類のサブプロトコルのヘッダおよびメッセージ列により1つのパケットが構成され）、それぞれのパケットの末尾には、そのサブプロトコルのFinishedメッセージが含まれる。

【 0 1 0 1 】

また、Neutralステートで使用されるサブプロトコル（Update Entity, Application Data Transfer, Operate Entity）によりサーバからクライアントに対して送信されるそれぞれのメッセージには、処理に失敗した場合に、それに続く後

続のメッセージの処理を続行するのか、または、中止するのかをクライアントに判断させる情報が含まれている。

【0 1 0 2】

例えば、クライアントは、先頭ビットが「0」であるメッセージの処理に失敗した場合、後続のメッセージの処理を中止し、一方、先頭ビットが「1」であるメッセージの処理に失敗した場合、後続のメッセージの処理を続行する。各サブプロトコルの拡張メッセージについても同様である。

【0 1 0 3】

アプリケーションのロジックが終了したとき、サーバは、ステップ S 5 において、Farewellサブプロトコルを使ってSERVER_GOOD_BYEメッセージをクライアントに送信する。SERVER_GOOD_BYEメッセージが送信されてから、コネクションが開放されるまでの状態がFarewellステートとされる（図 5 のstate3）。

【0 1 0 4】

サーバは、Farewell Protocolが終了したと判断したとき、ステップ S 6 において、クライアントに対してFINISHEDメッセージを送信し、コネクションを終了させる。クライアントにおいては、ステップ S 7 で、リソースが開放され、処理が終了される。

【0 1 0 5】

なお、SERVER_GOOD_BYE_DONEをサーバからクライアントへ送信することにより、Farewellステートを経ずに、コネクションが終了され、クライアントにおいて、リソースが開放されて、処理が終了する場合もある（図 2 5）。

【0 1 0 6】

また、任意の場所（状態）で処理を続行することが不可能なエラーが発生した場合、クライアントまたはサーバは、ステップ S 8 において、Errorサブプロトコルを用いてエラーメッセージを相手に送信する。エラーが発生してから制御プロトコルによる通信が終了されるまでの状態がErrorステートとされる（図 5 のstate4）。

【0 1 0 7】

ステップ S 9 において、例えば、サーバからFINISHEDメッセージがクライアン

トに送信されたとき、Errorステートが終了され、コネクションが終了される。

【0 1 0 8】

以上のような状態遷移を有する制御プロトコルに従って、クライアントとサーバ間で通信が行われる。

【0 1 0 9】

次に、Data Transfer Protocolにより規定されるパケットのデータフォーマットについて説明する。

【0 1 1 0】

図6は、パケットのデータフォーマットの例を示す図である。

【0 1 1 1】

図6に示されるように、パケットは、所定の数のメッセージ（サブプロトコル用の電文）からなるメッセージ列と、ヘッダから構成される。

【0 1 1 2】

ヘッダは、さらに、Version, Sub Protocol Type(SPT), Lengthの3つのフィールドから構成される。

【0 1 1 3】

Versionにより制御プロトコルのバージョン番号が表される。例えば、Versionのうちの上位バイトによりメジャーバージョン番号が表され、下位バイトによりマイナーバージョン番号が表される。Data Transfer Protocolの変更によりメジャーバージョン番号が上がり、サブプロトコルの追加または変更により、マイナーバージョン番号が上がる。

【0 1 1 4】

Sub Protocol TypeによりHandshake Protocol, Farewell Protocol, Error Protocol, Application Data Transfer Protocol, Update Entity Protocol, Operate Entity Protocolのうちの、パケットで用いられているサブプロトコルの種別が表される。

【0 1 1 5】

Lengthによりメッセージ列の長さが表される。

【0 1 1 6】

一方、メッセージ列に含まれる 1 つのメッセージは、Extension(Ext), Device ID(DID), Message Type(MT), Length, Dataのフィールドから構成される。

【0 1 1 7】

Extensionによりアプリケーションドメイン固有の処理を実行するためのエクステンションの種別が表される。

【0 1 1 8】

Device IDによりメッセージが対象とするデバイスのIDが表される。例えば、クライアントに複数のデバイスが接続されている場合、そのデバイス全体がメッセージの適用範囲とされるようなIDや、サーバ／クライアントとも、デバイスを意識することなくメッセージの適用範囲を決定するIDなども必要に応じて規定される。

【0 1 1 9】

Message Typeによりメッセージの種別が表される。後に詳述するように、例えば、Handshakeプロトコルには、クライアントからサーバに対して行われるCLIENT_HELLOメッセージや、サーバからクライアントに対して行われるSERVER_HELLOメッセージなどが規定されているが、それらのメッセージの種別がMessage Typeにより表される。

【0 1 2 0】

Lengthにより、メッセージのデータ部 (Data) の長さが表される。メッセージのデータ部のデータフォーマットはメッセージタイプに依存する。

【0 1 2 1】

次に、上述したそれぞれのサブプロトコルについて説明する。

【0 1 2 2】

図 7 は、Handshake Protocolにおいて用いられるメッセージを示す図である。

【0 1 2 3】

図 7 に示されるように、FINISHEDメッセージはサブプロトコルの終了を意味し、WARNINGメッセージは警告の通知を意味する。WARNINGメッセージのデータ部には、所定の文字列からなる警告メッセージが含まれる。

【0 1 2 4】

CLIENT_HELLOメッセージはクライアントからサーバに対するHandshakeデータの送信開始を意味し、CLIENT_HELLO_DONEメッセージはHandshakeデータの送信終了を意味する。

【 0 1 2 5 】

SERVER_HELLOメッセージはサーバからクライアントに対するHandshakeデータの送信開始を意味し、SERVER_HELLO_DONEメッセージはHandshakeデータの送信終了を意味する。

【 0 1 2 6 】

DEVICESメッセージはデバイスリストの通知を意味し、そのデータ部により、デバイス固有のデバイスID、パーミッション、タイプ名長、タイプ名、デバイス名長、およびデバイス名が表される。例えば、パーミッションによりデバイスに対するアクセス権の有無が表され、「0」がアクセス可（操作可）、「1」がアクセス不可（操作不可）とされる。

【 0 1 2 7 】

例えば、DEVICESメッセージにより、クライアントに接続されている、操作対象のデバイスリストの通知や、デバイスに実行させることが可能な処理の通知が行われる。

【 0 1 2 8 】

次に、図 8 のシーケンスを参照して、図 7 に示されるメッセージを用いて行われるHandshake Protocolについて説明する。以下のシーケンスにおいては、制御プロトコルによる通信によりデバイスを制御するサーバとしてのサーバ 2 と、クライアントとしての携帯電話機 5 により実行される処理について説明する。

【 0 1 2 9 】

Handshake Protocolは、ステップ S 2 1 において、例えば、携帯電話機 5（クライアント）のHandshakeメッセージ送信制御部 8 3（図 4）がCLIENT_HELLOメッセージを、例えば、サーバ 2（サーバ）に送信することにより開始される。すなわち、制御部 8 1 からの指示に応じて、Handshakeメッセージ管理部 8 2 においてCLIENT_HELLOメッセージが生成され、Handshakeメッセージ送信制御部 8 3 によりサーバ 2 に送信される。

【 0 1 3 0 】

携帯電話機 5 の Handshake メッセージ送信制御部 8 3 は、ステップ S 2 2 において、自分自身に接続されているデバイスであって、かつ、サーバ 2 によるコントロールを許可する全てのデバイスに関する情報を、DEVICES メッセージを用いてサーバに通知する。この DEVICES メッセージも、制御部 8 1 の指示に基づいて Handshake メッセージ管理部 8 2 により生成されたものである。

【 0 1 3 1 】

DEVICES メッセージにより、例えば、携帯電話機 5 に接続されているリーダライタ 1 1、およびリーダライタ 1 1 を介して接続されている外部 IC カード 1 2 に関する情報などがサーバ 2 に通知される。

【 0 1 3 2 】

アプリケーション固有のメッセージがステップ S 2 3 において送信され、ステップ S 2 4 において CLIENT_HELLO_DONE メッセージが送信されたとき、携帯電話機 5 からサーバ 2 に対する Handshake データの送信が終了される。

【 0 1 3 3 】

一方、携帯電話機 5 から送信されてきた CLIENT_HELLO メッセージ、DEVICES メッセージ、アプリケーション固有のメッセージ、および、CLIENT_HELLO_DONE メッセージがステップ S 4 1 乃至 S 4 4 においてサーバ 2 の Handshake メッセージ受信制御部 5 4 で取得されたとき、ステップ S 4 5 に進み、Handshake メッセージ送信制御部 5 3（図 3）は、携帯電話機 5 に対して SERVER_HELLO メッセージを送信する。

【 0 1 3 4 】

また、サーバ 2 の Handshake メッセージ送信制御部 5 3 は、ステップ S 4 6 において、アプリケーション固有のメッセージを送信し、ステップ S 4 7 に進み、SERVER_HELLO_DONE メッセージを送信する。ここでは、Handshake Protocol において送信される、アプリケーション固有のメッセージを含む全てのメッセージが Handshake メッセージ送信制御部 5 3 の制御の下に送信されている。

【 0 1 3 5 】

サーバ 2 から送信された SERVER_HELLO メッセージ、アプリケーション固有のメ

ッセージ、およびSERVER_HELLO_DONEメッセージは、ステップS 2 5乃至S 2 7において携帯電話機5のHandshakeメッセージ受信制御部8 4によりそれぞれ受信される。

【0 1 3 6】

サーバ2のHandshakeメッセージ送信制御部5 3は、携帯電話機5と自分自身の双方がHELLO_DONEメッセージ（CLIENT_HELLO_DONEメッセージ、SERVER_HELLO_DONEメッセージ）を送信したとき、ステップS 4 8において、さらに、携帯電話機5に対してFINISHEDメッセージを送信する。

【0 1 3 7】

サーバ2から送信されたFINISHEDメッセージは、ステップS 2 8において、携帯電話機5のHandshakeメッセージ受信制御部8 4により受信される。

【0 1 3 8】

携帯電話機5のHandshakeメッセージ送信制御部8 3は、ステップS 2 9においてアプリケーション固有のメッセージを送信し、ステップS 3 0に進み、サーバ2に対してFINISHEDメッセージを送信する。携帯電話機5から送信されたアプリケーション固有のメッセージとFINISHEDメッセージは、ステップS 4 9とS 5 0においてサーバ2のHandshakeメッセージ受信制御部5 4によりそれぞれ受信され、これにより携帯電話機5とサーバ2の双方においてHandshake Protocolが終了される。

【0 1 3 9】

なお、図8の一点鎖線は、アプリケーション固有のメッセージの送受信が必要な場合にのみ、それぞれ所定の数だけ送信されることを表している。例えば、ステップS 2 3において、メッセージが送信されない場合もあるし、複数のメッセージが送信される場合もある。後述する図1 0においても同様である。

【0 1 4 0】

また、図8においては、ステップS 5 0において、サーバ2によりFINISHEDメッセージが受信されたときにHandshake Protocolが終了されるとしたが、その前に、ステップS 4 4において、サーバ2で受信されたCLIENT_HELLO_DONEメッセージが受信されたときに、Handshake Protocolが終了されるようにしてもよい。

【0141】

さらに、図8においては、サーバ2と携帯電話機5から、それぞれHELLOメッセージとHELLO DONEメッセージが送信されとしたが、携帯電話機5からのみ、それらのメッセージがサーバ2に対して送信されるようにしてもよい。これにより、Handshake Protocolの確実性は多少損なわれるものの、Handshake Protocolを迅速に終了させ、Neutralステートにおける処理を迅速に開始させることができる。換言すれば、携帯電話機5とサーバ2の双方が_HELLOメッセージと_HELLO DONEメッセージを送受信することにより、より確実に、Handshake Protocolを実行させることができる。

【0142】

図9は、Farewell Protocolにおいて用いられるメッセージを示す図である。

【0143】

FINISHEDメッセージはサブプロトコルの終了を意味し、WARNINGメッセージは警告の通知を意味する。WARNINGメッセージのデータ部には、所定の文字列からなる警告メッセージが含まれる。

【0144】

CLIENT_GOOD_BYEメッセージは、クライアントからサーバに対するFarewellデータの送信開始を意味し、CLIENT_GOOD_BYE_DONEメッセージはFarewellデータの送信終了を意味する。

【0145】

一方、SERVER_GOOD_BYEメッセージは、サーバからクライアントに対するFarewellデータの送信開始を意味し、SERVER_GOOD_BYE_DONEメッセージはFarewellデータの送信終了を意味する。

【0146】

RETURN_CODEメッセージはサーバによる終了コードの通知を意味し、そのメッセージのデータ部には終了コードが含まれる。

【0147】

次に、図10のシーケンスを参照して、図9に示されるメッセージを用いて行われるFarewell Protocolについて説明する。

【 0 1 4 8 】

ステップ S 8 1 において、サーバ 2（サーバ）の Farewell メッセージ送信制御部 5 6 が SERVER_GOOD_BYE メッセージを送信することにより Farewell Protocol が開始される。このように、Farewell Protocol は、通信の主導権を有するサーバ 2 からのメッセージにより開始される（図 5 を参照して説明したように、Handshake Protocol が終了した後は、サーバが通信の主導権を有している）。

【 0 1 4 9 】

サーバ 2 の Farewell メッセージ送信制御部 5 6 は、ステップ S 8 2 において、携帯電話機 5（クライアント）に対して RETURN_CODE メッセージを送信し、ステップ S 8 3 に進み、アプリケーション固有のメッセージ（Extension 中の Farewell 用拡張メッセージ）を送信する。

【 0 1 5 0 】

また、サーバ 2 の Farewell メッセージ送信制御部 5 6 は、全てのアプリケーション固有のメッセージを送信した後、ステップ S 8 4 に進み、SERVER_GOOD_BYE_DONE メッセージを携帯電話機 5 に送信する。

【 0 1 5 1 】

ステップ S 8 1 乃至 S 8 4 においてサーバ 2 から送信された情報は、それぞれステップ S 6 1 乃至 S 6 4 において携帯電話機 5 の Farewell メッセージ受信制御部 8 7 により受信される。

【 0 1 5 2 】

携帯電話機 5 の Farewell メッセージ送信制御部 8 6 は、ステップ S 6 5 において、サーバ 2 に対して CLIENT_GOOD_BYE メッセージを送信する。

【 0 1 5 3 】

また、携帯電話機 5 の Farewell メッセージ送信制御部 8 6 は、ステップ S 6 6 において、アプリケーション固有のメッセージを送信し、ステップ S 6 7 に進み、CLIENT_GOOD_BYE_DONE メッセージを送信する。

【 0 1 5 4 】

携帯電話機 5 の Farewell メッセージ送信制御部 8 6 は、CLIENT_GOOD_BYE_DONE メッセージをサーバ 2 に対して送信した後、ステップ S 6 8 に進み、FINISHED メ

ッセージを送信する。ステップ S 6 5 乃至 S 6 8 において携帯電話機 5 の Farewell メッセージ送信制御部 8 6 から送信された情報は、それぞれステップ S 8 5 乃至 S 8 8 においてサーバ 2 の Farewell メッセージ受信制御部 5 7 により受信される。

【 0 1 5 5 】

サーバ 2 の Farewell メッセージ受信制御部 5 7 により、携帯電話機 5 から送信されてきた FINISHED メッセージがステップ S 8 8 において受信されたとき、ステップ S 8 9 に進み、Farewell メッセージ送信制御部 5 6 から携帯電話機 5 に対して FINISHED メッセージが送信され、Farewell Protocol が終了される。サーバ 2 の Farewell メッセージ送信制御部 5 6 から送信された FINISHED メッセージは、ステップ S 6 9 において携帯電話機 5 により受信される。

【 0 1 5 6 】

なお、ステップ S 8 4 において、SERVER_GOOD_BYE_DONE を送信することにより、Farewell Protocol が終了されるようにしてもよい。これにより、Farewell Protocol を迅速に終了させることができる。

【 0 1 5 7 】

図 1 1 は、Error Protocol において用いられるメッセージを示す図である。

【 0 1 5 8 】

FINISHED メッセージはサブプロトコルの終了を意味し、WARNING メッセージは警告の通知を意味する。WARNING メッセージのデータ部には、所定の文字列からなる警告メッセージが含まれる。

【 0 1 5 9 】

PACKET_FORMAT_ERROR メッセージはパケットのフォーマットにエラーがあることを意味し、ILLEGAL_STATE_ERROR メッセージは不正なメッセージを検知したことを意味する。

【 0 1 6 0 】

UNEXPECTED_ERROR メッセージは、予期しないエラーが発生したことを意味する。

【 0 1 6 1 】

PACKET_FORMAT_ERRORメッセージ、ILLEGAL_STATE_ERRORメッセージ、およびUNEXPECTED_ERRORメッセージのそれぞれのデータ部には、エラーの内容を表す、所定の文字列からなるエラーメッセージが含まれる。

【0 1 6 2】

次に、図 1 2 および図 1 3 のシーケンスを参照して、図 1 1 に示されるメッセージを用いて行われるError Protocolについて説明する。

【0 1 6 3】

図 1 2 は、携帯電話機 5 によりエラーが検出された場合のError Protocolのシーケンスを示す図である。

【0 1 6 4】

携帯電話機 5 の制御部 8 1 は、その後の処理を続行することが不可能なエラーを検知した場合、ステップ S 1 0 1 において、ERRORメッセージ管理部 8 8 を制御し、図 1 1 のメッセージ（PACKET_FORMAT_ERRORメッセージ、ILLEGAL_STATE_ERRORメッセージ、またはUNEXPECTED_ERRORメッセージ）を用いて、エラーを検知したこと（検知されたエラーの内容）をサーバ 2 に通知する。すなわち、制御部 8 1 からの指示に応じた所定のメッセージがERRORメッセージ管理部 8 8 により生成され、それがERRORメッセージ送信制御部 8 9 から送信される。

【0 1 6 5】

その後、携帯電話機 5 によるステップ S 1 0 2 および S 1 0 3、並びにサーバ 2 によるステップ S 1 1 2 および S 1 1 3 の処理により、携帯電話機 5 とサーバ 2 間でFINISHEDメッセージが送受信され、図 5 を参照して説明したように、サーバクライアント間の通信が終了される。

【0 1 6 6】

図 1 3 は、サーバ 2 によりエラーが検出された場合のError Protocolのシーケンスを示す図である。

【0 1 6 7】

サーバ 2 の制御部 5 1 は、その後の処理を続行することが不可能なエラーを検知した場合、ステップ S 1 3 1 において、ERRORメッセージ管理部 5 8 を制御し、検知したエラーの内容に応じたメッセージを用いて、エラーを検知したことを

携帯電話機 5 に通知する。すなわち、制御部 5 1 からの指示に応じた所定のメッセージが ERROR メッセージ管理部 5 8 により生成され、それが ERROR メッセージ送信制御部 5 9 から送信される。

【 0 1 6 8 】

サーバ 2 の ERROR メッセージ送信制御部 5 9 は、エラーを検知したことを通知したとき、ステップ S 1 3 2 に進み、携帯電話機 5 に対して FINISHED メッセージを送信する。

【 0 1 6 9 】

サーバ 2 から送信されたメッセージがステップ S 1 2 1 および S 1 2 2 において携帯電話機 5 により受信され、通信が終了される。

【 0 1 7 0 】

以上のように、サーバ 2 および携帯電話機 5 は、それぞれのサブプロトコルに基づく通信において、エラーを検知した場合、任意のタイミングで相手側に対して Error メッセージを送信することができる。

【 0 1 7 1 】

なお、上述した Error サブプロトコルによるメッセージだけでなく、アプリケーションにより独自に規定されたエラー通知メッセージが用いられて、相手側にエラーの検知が通知されるようにしてもよい。

【 0 1 7 2 】

また、Error メッセージのうちの WARNING メッセージは、シーケンスに影響を与えない。すなわち、WARNING メッセージが受信された側において、メッセージの内容がログに出力されるようにしてもよいし、無視されるようにしてもよい。

【 0 1 7 3 】

図 1 4 は、Update Entity Protocol において用いられるメッセージを示す図である。

【 0 1 7 4 】

Update Entity Protocol のメッセージは、例えば、図 1 のリーダライタ 1 1 の表面に設けられている LED (Light Emitting Diode) を発光させたり、または、所定のデバイスの電源をオンにするなど、デバイスの状態（属性）を変更させると

きに用いられる。

【0175】

FINISHEDメッセージはパケットの終端を意味し、WARNINGメッセージは警告の通知を意味する。WARNINGメッセージのデータ部には所定の文字列からなる警告メッセージが含まれる。

【0176】

SET_PROPERTYメッセージは、サーバからクライアントに対して送信され、クライアントに接続されているデバイスの属性値の設定を意味する。SET_PROPERTYメッセージに対するクライアントからの返信は、PROPERTY_SETメッセージにより行われる。

【0177】

このSET_PROPERTYメッセージは、メッセージに基づく処理がクライアントにおいて失敗したとき、後続のメッセージの処理を中止しなければならないメッセージである。従って、SET_PROPERTYメッセージの先頭ビットは、後続のメッセージの処理を中断するように設定される。

【0178】

SET_PROPERTYメッセージのデータ部には、属性名長、属性名、属性値長、属性値などの、デバイスに新たに設定させる属性に関する情報が含まれる。

【0179】

PROPERTY_SETメッセージは、サーバから送信されてきたSET_PROPERTYメッセージに対するクライアントからの返信を意味し、そのデータ部には、SET_PROPERTYメッセージに基づく処理がクライアントにおいて成功したのか否かを表すフラグが含まれる。

【0180】

GET_PROPERTYメッセージは、サーバからクライアントに対して送信され、クライアントに接続されたデバイスの属性値の取得を意味する。GET_PROPERTYメッセージに対するクライアントからの返信は、PROPERTYメッセージにより行われる。

【0181】

このGET_PROPERTYメッセージは、メッセージに基づく処理がクライアントにお

いて失敗したとき、後続のメッセージの処理を中止しなければならないメッセージである。従って、GET_PROPERTYメッセージの先頭ビットは、後続のメッセージの処理を中断するように設定される。

【0 1 8 2】

GET_PROPERTYメッセージのデータ部には、属性値の取得を要求するデバイスの属性名長、および属性名などの情報が含まれる。

【0 1 8 3】

PROPERTYメッセージは、サーバから送信されてきたGET_PROPERTYメッセージに対するクライアントからの返信を意味し、そのデータ部には、サーバからの要求に対応する、デバイスの属性値長および属性値などの情報が含まれる。

【0 1 8 4】

SET_NETWORK_TIMEOUTメッセージは、クライアントのネットワークタイムアウト値の設定を意味する。

【0 1 8 5】

このSET_NETWORK_TIMEOUTメッセージは、メッセージに基づく処理がクライアントにおいて失敗した場合であっても、後続のメッセージの処理を継続するメッセージである。従って、SET_NETWORK_TIMEOUTメッセージの先頭ビットは、後続のメッセージの処理を継続するように設定される。

【0 1 8 6】

SET_NETWORK_TIMEOUTメッセージのデータ部には、クライアントに設定させるタイムアウト値が含まれる。

【0 1 8 7】

次に、図 1 5 のシーケンスを参照して、図 1 4 に示されるメッセージを用いて行われるUpdate Entity Protocolについて説明する。

【0 1 8 8】

携帯電話機 5 に接続されたデバイスの状態を変更するとき、サーバ 2 のUpdate Entityメッセージ管理部 6 4 は、制御部 5 1 による制御に基づいて、ステップ S 1 5 1 において、携帯電話機 5 に送信するUpdate Entityメッセージを生成する。生成されたUpdate EntityメッセージはUpdate Entityメッセージ送信制御部

65に出力される。

【0189】

サーバ2のUpdate Entityメッセージ送信制御部65は、ステップS152において、ステップS151で生成されたUpdate Entityメッセージ列の末尾にFINISHEDメッセージを付加し、Update Entityメッセージ列とFINISHEDメッセージからなるパケットを携帯電話機5に送信する。

【0190】

すなわち、ステップS151で生成される所定の数のUpdate Entityメッセージと、ステップS152で付加される1つのFINISHEDメッセージからなる1つのパケットが携帯電話機5に対して送信される。

【0191】

このように、1つのパケットに複数のメッセージを含めて携帯電話機5に送信することができるため（複数のメッセージを送信するのに1つのヘッダやフッタを付加するだけでよい）、1つのメッセージ毎にヘッダやフッタが付加されて構成されるパケットが送信される場合に較べて、データ量を抑制することができる。また、通信時間を抑制できるとともに、通信コストを抑制することができる。このことは、全てのプロトコルにおいても同様である。

【0192】

パケットに含められてサーバ2から送信されたUpdate Entityメッセージは、ステップS141において携帯電話機5のUpdate Entityメッセージ受信制御部96により取得され、Update Entityメッセージ列の末尾に付加されたFINISHEDメッセージは、ステップS142において取得される。

【0193】

携帯電話機5においては、サーバから送信されてきたパケットに含まれるメッセージが取得され、取得された順に、メッセージに基づく処理が行われる。また、必要に応じて、処理結果を表すレスポンスが制御部81により生成される。

【0194】

携帯電話機5のUpdate Entityメッセージ受信制御部96は、サーバ2から送信されてきたFINISHEDメッセージを受信することによりパケットの終端を検知す

る。受信したUpdate Entityメッセージに対するレスポンス（Update Entityメッセージ）がステップS 1 4 3においてUpdate Entityメッセージ管理部 9 4 により生成される。

【 0 1 9 5 】

例えば、サーバ2 からデバイスの属性値を変更することが指示されたとき、属性値の変更がデバイス制御部 1 0 0 により行われ、属性値の変更が成功したか否かが制御部 8 1 からUpdate Entityメッセージ管理部 9 4 に通知される。Update Entityメッセージ管理部 9 4 においては、その通知に対応するレスポンスが生成され、Update Entityメッセージ送信制御部 6 5 に出力される。

【 0 1 9 6 】

ステップS 1 4 4 において、携帯電話機5 のUpdate Entityメッセージ送信制御部 9 5 は、生成したレスポンスにFISNISHEDメッセージを付加し、1つのパケットとしてサーバ2 に送信する。なお、レスポンスの有無、およびその数は、サーバ2 から送信されてきたメッセージ次第である。

【 0 1 9 7 】

パケットに含められて携帯電話機5 から送信されたUpdate Entityメッセージは、ステップS 1 5 3 においてサーバ2 のUpdate Entityメッセージ受信制御部 6 6 により取得され、Update Entityメッセージ列の末尾に付加されたFINISHEDメッセージは、ステップS 1 5 4 において取得される。

【 0 1 9 8 】

以下、適宜、Neutralステートにおいて、サーバ2 から携帯電話機5 に対して送信されるパケット（0を含む所定の数のUpdate Entityメッセージと1つのFISNISHEDメッセージを含むパケット、0を含む所定の数のApplication Data Transferメッセージと1つのFISNISHEDメッセージを含むパケット、または、0を含む所定の数のOperate Entityメッセージと1つのFISNISHEDメッセージを含むパケット）を制御パケットと称する。

【 0 1 9 9 】

また、その制御パケットに含まれるメッセージの処理結果等を通知すべく、携帯電話機5 からサーバ2 に対して返信されるパケット（メッセージに対応する0

を含む所定の数のレスポンスと1つのFISNISHEDメッセージを含むパケット)を返信パケットと称する。

【0200】

以上のようにして、Update EntityメッセージとFISNISHEDメッセージを含む制御パケットが、サーバ主導でサーバ2から携帯電話機5に対して送信され、それに基づく処理が携帯電話機5において実行される。携帯電話機5においては、サーバ2から送信されてきたUpdate Entityメッセージに対応するレスポンスが生成され、レスポンスとFISNISHEDメッセージを含む返信パケットがサーバ2に対して送信される。

【0201】

具体的には、例えば、図14のSET_PROPERTYメッセージを含む制御パケットがサーバ2から携帯電話機5に対して送信されたとき、携帯電話機5においては、SET_PROPERTYメッセージに基づくデバイスの属性値の設定が行われ、PROPERTY_SETメッセージ(レスポンス)を含む返信パケットがサーバ2に対して返信される。

【0202】

なお、携帯電話機5およびサーバ2は、Update Entityメッセージ列の中に、所定の数のアプリケーション固有メッセージ(Extension中のUpdate Entity用拡張メッセージ)を含めて送信することができる。

【0203】

図16は、Application Data Transfer Protocolにおいて用いられるメッセージを示す図である。

【0204】

Application Data Transfer Protocolは、例えば、リーダライタ11を介して携帯電話機5に接続される外部ICカード12を制御するためのアプリケーションのコマンドなど、サーバとクライアント(デバイス)間におけるアプリケーションデータを送受信するときに用いられる。

【0205】

FINISHEDメッセージはパケットの終端を意味し、WARNINGメッセージは警告の

通知を意味する。WARNINGメッセージのデータ部には所定の文字列からなる警告メッセージが含まれる。

【0 2 0 6】

次に、図 1 7 のシーケンスを参照して、図 1 6 に示されるメッセージを用いて行われるApplication Data Transfer Protocolについて説明する。

【0 2 0 7】

ステップ S 1 7 1 において、サーバ 2 のApplication Data Transferメッセージ管理部 6 1 は、一群のApplication Data Transferメッセージ列を生成する。生成されたApplication Data Transferメッセージは、Application Data Transferメッセージ送信制御部 6 2 に出力される。

【0 2 0 8】

サーバ 2 のApplication Data Transferメッセージ送信制御部 6 2 は、ステップ S 1 7 2 において、生成したApplication Data Transferメッセージ列の末尾にFINISHEDメッセージを付加し、所定の数のApplication Data Transferメッセージ列と 1 つのFINISHEDメッセージからなる制御パケットを携帯電話機 5 に送信する。

【0 2 0 9】

制御パケットに含まれるApplication Data Transferメッセージは、ステップ S 1 6 1 において、携帯電話機 5 のApplication Data Transferメッセージ受信制御部 9 3 により受信され、FINISHEDメッセージはステップ S 1 6 2 において受信される。

【0 2 1 0】

携帯電話機 5 のApplication Data Transferメッセージ受信制御部 9 3 は、サーバ 2 から送信されてきたFINISHEDメッセージを受信することにより制御パケットの終端を検知し、制御パケットに含まれるApplication Data TransferメッセージがApplication Data Transferメッセージ管理部 9 1 に出力される。

【0 2 1 1】

ステップ S 1 6 3 において、パケットに含まれるApplication Data Transferメッセージに対応するレスポンスが生成される。すなわち、デバイス制御部 1 0

0 により行われた制御の結果がApplication Data Transferメッセージ管理部 9 1 に通知され、Application Data Transferメッセージ管理部 9 1 により、その結果に対応するレスポンスが生成される。生成されたレスポンスはApplication Data Transferメッセージ送信制御部 9 2 に供給される。

【 0 2 1 2 】

ステップ S 1 6 4 において、携帯電話機 5 のApplication Data Transferメッセージ送信制御部 9 2 は、生成されたレスポンス（Application Data Transferメッセージ）列の末尾にFINISHEDメッセージを付加し、生成した返信パケットをサーバ 2 に送信する。

【 0 2 1 3 】

パケットに含められて携帯電話機 5 から送信されたApplication Data Transferメッセージは、ステップ S 1 7 3 においてサーバ 2 のApplication Data Transferメッセージ受信制御部 6 3 により取得され、Application Data Transferメッセージ列の末尾に付加されたFINISHEDメッセージは、ステップ S 1 7 4 において取得される。

【 0 2 1 4 】

なお、レスポンスの有無、およびその数は、サーバ 2 から送信されてきたメッセージ次第である。

【 0 2 1 5 】

上述したUpdate Entity Protocolの場合と同様に、携帯電話機 5 およびサーバ 2 は、Application Data Transferメッセージ列の中に、所定の数 of アプリケーション固有メッセージ（Extension中のApplication Data Transfer用拡張メッセージ）を送信することができる。

【 0 2 1 6 】

図 1 8 は、Operate Entity Protocolにおいて用いられるメッセージを示す図である。

【 0 2 1 7 】

Operate Entity Protocolは、例えば、携帯電話機 5 に接続されたリーダライタ 1 1 による電磁波の輻射の開始または停止、或いは、ユーザに対して所定の入

力を行わせるための所定の画面表示など、クライアントに接続されたデバイスを制御するときに用いられる。

【 0 2 1 8 】

FINISHEDメッセージはパケットの終端を意味し、WARNINGメッセージは警告の通知を意味する。WARNINGメッセージのデータ部には、所定の文字列からなる警告メッセージが含まれる。

【 0 2 1 9 】

UPDATE_VIEWメッセージは、サーバからクライアントに対して送信され、クライアントにおけるビュー（表示画面）の更新を意味する。UPDATE_VIEWメッセージに対する返信は、VIEW_UPDATEDメッセージによりサーバに対して行われる。

【 0 2 2 0 】

このUPDATE_VIEWメッセージは、メッセージに基づく処理がクライアントにおいて失敗したとき、後続のメッセージの処理を中止しなければならないメッセージである。従って、UPDATE_VIEWメッセージの先頭ビットは、後続のメッセージの処理を中断するように設定される。

【 0 2 2 1 】

VIEW_UPDATEDメッセージは、サーバから送信されてきたUPDATE_VIEWメッセージに対する返信を意味し、そのデータ部には、ビューの更新が成功したか否かを表すフラグが含まれる。

【 0 2 2 2 】

BEGIN_DATA_INPUTメッセージは、サーバからクライアントに対して送信され、デバイスに対するデータの入力を開始することを意味する。BEGIN_DATA_INPUTメッセージに対するクライアントからの返信は、END_DATA_INPUTによりサーバに対して行われる。

【 0 2 2 3 】

このBEGIN_DATA_INPUTメッセージは、メッセージに基づく処理がクライアントにおいて失敗したとき、後続のメッセージの処理を中止しなければならないメッセージである。従って、BEGIN_DATA_INPUTメッセージの先頭ビットは、その処理に失敗した場合、後続のメッセージの処理を中断するように設定される。

【 0 2 2 4 】

END_DATA_INPUTメッセージは、BEGIN_DATA_INPUTに対するクライアントからの返信を意味し、そのデータ部には、データの入力に成功したか否かを表すフラグ、属性名長、属性名、属性値長、および属性値の情報が含まれる。

【 0 2 2 5 】

OPERATE_DEVICEメッセージは、サーバからクライアントに対して送信され、クライアントに接続されたデバイスを実行することを意味する。OPERATE_DEVICEメッセージに対するクライアントからの返信は、DEVICE_RESPONSEメッセージによりサーバに対して行われる。

【 0 2 2 6 】

このOPERATE_DEVICEメッセージは、メッセージに基づく処理がクライアントにおいて失敗したとき、後続のメッセージの処理を中止しなければならないメッセージである。従って、OPERATE_DEVICEメッセージの先頭ビットは、その処理に失敗した場合、後続のメッセージの処理を中断するように設定される。

【 0 2 2 7 】

OPERATE_DEVICEメッセージのデータ部には、操作名文字数、操作名、パラメータ長、およびパラメータの情報が含まれる。

【 0 2 2 8 】

DEVICE_RESPONSEメッセージは、クライアントからサーバに対して送信され、デバイス操作の結果の通知を意味する。DEVICE_RESPONSEメッセージのデータ部には、レスポンス長、レスポンスの情報が含まれる。

【 0 2 2 9 】

PLAY_SOUNDメッセージは、サーバからクライアントに対して送信され、クライアント側において、サーバにより指定された論理名に対応した音を発生させることを意味する。PLAY_SOUNDメッセージに対する返信は行われない。

【 0 2 3 0 】

このPLAY_SOUNDメッセージは、メッセージに基づく処理がクライアントにおいて失敗した場合であっても、後続のメッセージの処理を継続するメッセージである。従って、PLAY_SOUNDメッセージの先頭ビットは、その処理に失敗した場合で

あっても、後続のメッセージの処理を継続するように設定される。

【0231】

PLAY_SOUNDメッセージのデータ部には、発生させる音の論理名を表す情報が含まれる。

【0232】

次に、図19のシーケンスを参照して、図18に示されるメッセージを用いて行われるOperate Entity Protocolについて説明する。

【0233】

サーバ2のOperate Entityメッセージ管理部67は、ステップS191において、携帯電話機5に送信する、意味のある一群のOperate Entityメッセージ列を生成する。生成されたOperate EntityメッセージはOperate Entityメッセージ送信制御部68に出力される。

【0234】

携帯電話機5のOperate Entityメッセージ送信制御部68は、ステップS192において、生成したメッセージ列の末尾にFINISHEDメッセージを付加し、制御パケットとして携帯電話機5に送信する。

【0235】

携帯電話機5に対して送信された制御パケットに含まれるOperate EntityメッセージおよびFINISHEDメッセージは、ステップS181およびS182においてそれぞれ受信される。

【0236】

携帯電話機5のOperate Entityメッセージ管理部97は、ステップS183において、サーバ2から送信されてきたOperate Entityメッセージに対するレスポンスを制御部81による指示に基づいて生成する。なお、レスポンスの有無、およびその数は、サーバ2から送信されてきたメッセージ次第である。

【0237】

携帯電話機5のOperate Entityメッセージ送信制御部65は、ステップS184において、Operate Entityメッセージ管理部97により生成されたレスポンスにFINISHEDメッセージを付加し、返信パケットとしてサーバ2に送信する。

【 0 2 3 8 】

パケットに含められて携帯電話機 5 から送信された Operate Entity メッセージは、ステップ S 1 9 3 においてサーバ 2 の Operate Entity メッセージ受信制御部 6 9 により取得され、Operate Entity メッセージ列の末尾に付加された FINISHED メッセージは、ステップ S 1 9 4 において取得される。

【 0 2 3 9 】

なお、携帯電話機 5、およびサーバ 2 は、Operate Entity メッセージ列の中に所定の数のアプリケーション固有メッセージ（Extension 中の Operate Entity 用拡張メッセージ）を送信することができる。

【 0 2 4 0 】

以上のように、携帯電話機 5 とサーバ 2 の間で Handshake が成立し、Neutral ステート（図 5）のときには、サーバ 2 が通信の主導権を有し、サーバ 2 から、Update Entity Protocol、Application Data Transfer Protocol、または Operate Entity Protocol に基づくメッセージが携帯電話機 5 に対して送信される（Update Entity Protocol による通信のシーケンスを示す図 1 5、Application Data Transfer Protocol による通信のシーケンスを示す図 1 7、および、Operate Entity Protocol による通信のシーケンスを示す図 1 9 においては、いずれもサーバ 2 から通信が開始されている）。

【 0 2 4 1 】

すなわち、携帯電話機 5 からの命令に応じて、サーバ 2 がレスポンスを携帯電話機 5 に返信し、その返信に基づいてデバイスが制御されるのではなく、サーバ 2 から携帯電話機 5 に対して直接送信された命令に基づいて、携帯電話機 5 に接続されているデバイスが制御される。

【 0 2 4 2 】

従って、1 度通信が確立された後は、サーバ 2 からの通信によりデバイスを直接制御することができる。

【 0 2 4 3 】

次に、図 2 0 を参照して、以上のようなサブプロトコルに従って Neutral ステートにおける通信を行う、サーバ（サーバ 2）とクライアント（携帯電話機 5）

の処理について説明する。

【0 2 4 4】

図 2 0 においては、ステップ S 2 1 1 において、Application Data Transfer メッセージ A (MsgA)、Application Data Transfer メッセージ B (MsgB) に Finished メッセージが付加されて生成された制御パケット (Application Data Transfer パケット) 2 0 1 がサーバ 2 から携帯電話機 5 に対して送信されている。

【0 2 4 5】

図 1 7 を参照して説明したように、Application Data Transfer Protocol による通信においては、サーバ 2 の Application Data Transfer メッセージ送信制御部 6 2 から、所定の数の Application Data Transfer メッセージに Finished メッセージが付加されて構成される制御パケットが携帯電話機 5 に対して送信される。

【0 2 4 6】

また、ステップ S 2 1 1 においては、Operate Entity メッセージ X (MsgX) に Finished メッセージが付加されて生成された制御パケット (Operate Entity パケット) 2 0 2 がサーバ 2 から携帯電話機 5 に対して送信されている。

【0 2 4 7】

図 1 9 を参照して説明したように、Operate Entity Protocol による通信においては、サーバ 2 から、所定の数の Operate Entity メッセージに Finished メッセージが付加されて構成される制御パケットが携帯電話機 5 に対して送信される。

【0 2 4 8】

さらに、ステップ S 2 1 1 においては、Application Data Transfer メッセージ C (MsgC) に Finished メッセージが付加されて生成された制御パケット (Application Data Transfer パケット) 2 0 3 がサーバ 2 から携帯電話機 5 に対して送信されている。

【0 2 4 9】

このように、複数の制御パケットが携帯電話機 5 に一括して送信されるため、それぞれのプロトコルのパケットをそれぞれ送信する場合に較べて、通信回数を抑制することができる。

【 0 2 5 0 】

制御パケットに含められてサーバ 2 から送信されたメッセージは、送信された順序で携帯電話機 5 により処理され、レスポンスが順次生成される。携帯電話機 5 により生成されたレスポンスは、生成された順序でサーバ 2 に返信される。

【 0 2 5 1 】

従って、図 2 0 の例においては、ステップ S 2 0 1 で携帯電話機 5 により制御パケット 2 0 1 乃至 2 0 3 が受信されたとき、Application Data Transfer メッセージ A、Application Data Transfer メッセージ B、Operate Entity メッセージ X、Application Data Transfer メッセージ C の順に、それらのメッセージに基づく処理が携帯電話機 5 により実行される。

【 0 2 5 2 】

上述したように、メッセージに基づく処理の途中でエラーが発生した場合、それに続くメッセージの処理を続行するのか、または、その時点で処理を中断するのかは、メッセージの先頭ビットの値により設定されている。

【 0 2 5 3 】

図 2 0 に示される処理においては、送信された全てのメッセージの処理が成功した場合の例とされている。また、Application Data Transfer メッセージ A と Operate Entity メッセージ X は、それぞれ返信（レスポンス）が必要とされるメッセージであり、Application Data Transfer メッセージ B と Application Data Transfer メッセージ C は、共に返信が必要でないメッセージとされている。

【 0 2 5 4 】

携帯電話機 5 において処理が成功したか否かをサーバ 2 側で管理する必要があるメッセージに対しては、携帯電話機 5 によりレスポンスが生成される。

【 0 2 5 5 】

従って、携帯電話機 5 は、Application Data Transfer メッセージ A に基づく処理を実行したとき、それに対するレスポンス A（ResA）を生成し、レスポンス A に Finished メッセージを付加して生成した返信パケット（Application Data Transfer パケット） 2 1 1 を、ステップ S 2 0 2 において、サーバ 2 に対して送信する。

【 0 2 5 6 】

図 1 7 を参照して説明したように、Application Data Transfer Protocol による通信においては、サーバ 2 から送信されてきた制御パケットに含まれる Application Data Transfer メッセージに対応して Application Data Transfer メッセージ管理部 9 2 によりレスポンスが生成され、生成されたレスポンスと Finished メッセージからなる返信パケットが Application Data Transfer メッセージ送信制御部 9 2 からサーバ 2 に送信される。

【 0 2 5 7 】

また、携帯電話機 5 は、Operate Entity メッセージ X に基づく処理を実行したとき、それに対するレスポンス X (Res X) を生成し、レスポンス X に Finished メッセージを付加して生成した返信パケット (Operate Entity パケット) 2 1 2 を、ステップ S 2 0 2 において、サーバ 2 に対して送信する。

【 0 2 5 8 】

図 1 9 を参照して説明したように、Operate Entity Protocol による通信においては、サーバ 2 から送信されてきた制御パケットに含まれる Operate Entity メッセージに対応して、Operate Entity メッセージ管理部 9 7 によりレスポンスが生成され、生成されたレスポンスと Finished メッセージからなる返信パケットが Operate Entity メッセージ送信制御部 9 8 からサーバ 2 に送信される。

【 0 2 5 9 】

なお、携帯電話機 5 は、返信メッセージの有無に関わらず、サーバ 2 から送信されてきた 1 つの制御パケットに対して 1 つの返信パケットをサーバ 2 に送信しなければならないため、Application Data Transfer メッセージ C が返信を必要としないメッセージである場合であっても、制御パケット 2 0 3 に対して、Finished メッセージが末尾に付加された返信パケット (Application Data Transfer パケット) 2 1 3 をステップ S 2 0 2 においてサーバ 2 に送信する。

【 0 2 6 0 】

携帯電話機 5 から送信された返信パケットはステップ S 2 1 2 においてサーバ 2 により受信され、返信パケットに含まれるレスポンスにより、携帯電話機 5 においてメッセージに基づく処理が成功されたことが取得される。

【 0 2 6 1 】

このように、サーバ 2 から送信されたそれぞれのメッセージに対して、送信された順序で返信メッセージが生成されるため、サーバ 2 および携帯電話機 5 は、送信メッセージと返信メッセージを対応付けて管理することができる。

【 0 2 6 2 】

図 2 1 は、サーバ 2 と携帯電話機 5 により実行される他の処理について説明するシーケンスである。

【 0 2 6 3 】

図 2 1 においては、ステップ S 2 3 1 において、図 2 0 の場合と同様に、制御パケット 2 2 1 乃至 2 2 3 がサーバ 2 から携帯電話機 5 に対して送信されている。

【 0 2 6 4 】

なお、図 2 1 においては、携帯電話機 5 に対して送信された Application Data Transfer メッセージ A (MsgA) の処理が失敗した場合の例とされている。また、Application Data Transfer メッセージ A の先頭ビットにより、Application Data Transfer メッセージ A の処理が失敗した場合、その時点で処理を中断することが表されている場合の例とされている。

【 0 2 6 5 】

従って、ステップ S 2 2 1 において、携帯電話機 5 により制御パケット 2 2 1 に含まれる Application Data Transfer メッセージ A が取得され、それに基づく処理が失敗したとき、Application Data Transfer メッセージ A に続く Application Data Transfer メッセージ B、Operate Entity メッセージ X、Application Data Transfer メッセージ C に基づく処理は実行されない。

【 0 2 6 6 】

携帯電話機 5 は、Application Data Transfer メッセージ A に基づく処理に失敗したとき、処理が失敗したことをサーバ 2 に通知するレスポンス A (ResA) を生成する。携帯電話機 5 は、ステップ S 2 2 2 において、生成したレスポンス A に Finished メッセージを付加し、返信パケット (Application Data Transfer パケット) 2 3 1 をサーバ 2 に送信する。

【0267】

ステップS232において、サーバ2は、携帯電話機5からの返信パケット231を受信し、それに含まれるレスポンスAに基づいて、Application Data TransferメッセージAに基づく処理が携帯電話機5において失敗したことを取得する。また、サーバ2は、Application Data TransferメッセージB、Operate EntityメッセージX、Application Data TransferメッセージCに基づく処理が実行されていないことを取得する。

【0268】

図22は、サーバ2と携帯電話機5により実行されるさらに他の処理について説明するシーケンスである。

【0269】

図22においても、図20または図21と同様に、ステップS251において、制御パケット241乃至242がサーバ2から携帯電話機5に対して送信されている。

【0270】

なお、図22においては、携帯電話機5に対して送信されたApplication Data TransferメッセージB (MsgB) の処理が失敗した場合の例とされている。また、Application Data TransferメッセージBの先頭ビットにより、Application Data TransferメッセージBの処理が失敗した場合であっても、それに続くメッセージの処理を続行することが表されている場合の例とされている。

【0271】

さらに、Application Data TransferメッセージAおよびOperate EntityメッセージXは、返信メッセージを必要とするメッセージであり、Application Data TransferメッセージCは、返信メッセージを必要としないメッセージとされている。

【0272】

従って、携帯電話機5は、ステップS241において、制御パケット241を受信し、Application Data TransferメッセージAに基づく処理を実行し、レスポンスA (ResA) を生成する。

【 0 2 7 3 】

また、携帯電話機 5 は、Application Data Transfer メッセージ B に基づく処理に失敗した場合であっても、Application Data Transfer メッセージ B の先頭ビットにより表される指示に従って、それに続く処理を続行する。すなわち、携帯電話機 5 は、制御パケット 2 4 2 に含まれる Operate Entity メッセージ X に基づく処理を実行し、Operate Entity メッセージ X に対応するレスポンス X (Res X) を生成する。

【 0 2 7 4 】

さらに、携帯電話機 5 は、レスポンス X を生成した後、制御パケット 2 4 3 に含まれる Application Data Transfer メッセージ C に基づく処理を実行する。

【 0 2 7 5 】

携帯電話機 5 により生成されたレスポンス A と Finished パケットを含む返信パケット (Application Data Transfer パケット) 2 5 1、レスポンス X と Finished パケットを含む返信パケット (Operate Entity パケット) 2 5 2、および、Finished パケットを含む返信パケット (Application Data Transfer パケット) 2 5 3 は、ステップ S 2 4 2 において、サーバ 2 に対して送信される。

【 0 2 7 6 】

携帯電話機 5 から送信された返信パケットは、ステップ S 2 5 2 においてサーバ 2 により受信され、Application Data Transfer メッセージ A に基づく処理、および、Operate Entity メッセージに基づく処理が携帯電話機 5 において成功したこと、並びに、Application Data Transfer メッセージ B に基づく処理が失敗したことが取得され、処理が終了される。

【 0 2 7 7 】

Neutral ステートのとき、以上のようなサーバ 2 と携帯電話機 5 の間でのパケットの送受信が繰り返し実行され、サーバ主導によるクライアント (デバイス) の制御が行われる。

【 0 2 7 8 】

以上においては、携帯電話機 5 に対するメッセージの送信と、サーバ 2 に対するレスポンスの返信が繰り返されることにより、所定のデバイスの制御が行われ

るとしたが、例えば、サーバ2から携帯電話機5に対してスクリプト（簡易プログラム）が制御パケットと共に一括して送信され、メッセージに基づく処理結果と、簡易プログラムに従って、次に実行する処理が携帯電話機5自身により判断されるようにしてもよい。

【0279】

すなわち、スクリプトには、所定のメッセージに対して生成されたレスポンス（デバイスのレスポンス）の内容によって、次に、どのメッセージに基づく処理を実行するのかが記述される。

【0280】

スクリプトに基づく処理が終了されたとき、デバイスから携帯電話機5に対して行われたレスポンスの内容が、サーバ2に対して一括して送信される返信パケットに含まれて送信される。

【0281】

スクリプトの実行履歴（携帯電話機5からデバイスに送信されたメッセージ（携帯電話機5により実行されたメッセージ）の内容と、その順序を表す情報）がサーバ2に送信されることにより、サーバ2は、携帯電話機5から送信されてきたレスポンスが、どのメッセージに対するものなのかを判断することができる。

【0282】

また、以上においては、Handshakeプロトコルにより通信が確立したとき、サーバ2が通信の主導となり、携帯電話機5との間での通信が行われるとしたが、任意のタイミングで、携帯電話機5が通信の主導権を有することができるようにしてもよい。

【0283】

この場合、例えば、主導権の逆転は、サーバ・クライアント送受信方向逆転プロトコルによって行われる。

【0284】

例えば、サーバ2が主導となって通信が行われている場合、所定のタイミングで、サーバ2から携帯電話機5に対してサーバ・クライアント送受信方向逆転プロトコルのEXCHANGEメッセージが送信される。それ以降の通信においては、携帯

電話機 5 が主導となって、上述したものと同様の通信が行われる。

【 0 2 8 5 】

一方、携帯電話機 5 が主導となって通信が行われている場合、所定のタイミングで、携帯電話機 5 からサーバ 2 に対してサーバ・クライアント送受信方向逆転プロトコル（EXCHANGE プロトコル）の EXCHANGE メッセージが送信される。この場合、サーバ 2 が主導となり、それ以降の通信が行われる。

【 0 2 8 6 】

図 2 3 および図 2 4 は、EXCHANGE（交換）プロトコルのシーケンスを示す図である。

【 0 2 8 7 】

図 2 3 においては、始めに、携帯電話機 5（Client）が通信の主導を有している。ステップ S 3 0 1 において携帯電話機 5 から送信された EXCHANGE メッセージは、ステップ S 3 1 1 においてサーバ 2（Server）により受信され、それ以降、サーバ 2 が主導となり、サーバ 2 と携帯電話機 5 との間で通信が行われる。

【 0 2 8 8 】

一方、図 2 4 においては、始めに、サーバ 2 が通信の主導を有している。ステップ S 3 3 1 においてサーバ 2 から送信された EXCHANGE メッセージは、ステップ S 3 2 1 において携帯電話機 5 により受信され、それ以降、携帯電話機 5 が主導となり、サーバ 2 と携帯電話機 5 との間で通信が行われる。

【 0 2 8 9 】

以上のようにして、Neutral ステートでの通信の主導権の切り替えが行われる。

【 0 2 9 0 】

なお、サーバ 2 と携帯電話機 5 の間で確立された通信経路上に携帯電話機 5 のファイアウォールが存在する場合、そのファイアウォールを越えて通信を行うことができるようにして、サーバ 2 と携帯電話機 5 の間では、HTTP による通信が行われる。この場合、上述した複数のプロトコルのパケットの送受信は、1 回のセッションが維持された状態で行われる。

【 0 2 9 1 】

図 2 5 は、制御プロトコルの状態遷移について説明する他の図であり、基本的には、図 5 を参照して説明したものと同様である。上述した図 5 に代えて、図 2 5 に示されるように、状態遷移が行われるようにしてもよい。なお、図 5 と重複する部分については、その説明を適宜省略する。

【 0 2 9 2 】

クライアントは、Handshake Protocol (State1) を終了したとき、ステップ S 3 4 3 において、サーバに対して CLIENT_HELLO_DONE メッセージを送信する。これによりコネクションが確立され、制御の主導権がサーバに移る。

【 0 2 9 3 】

また、Neutral ステート状態にあるステップ S 3 4 4 においては、サーバが主導となり、Update Entity サブプロトコル、Application Data Transfer サブプロトコル、および、Operate Entity サブプロトコルによる通信が行われるだけでなく、必要に応じて、上述した EXCHANGE メッセージが用いられ、主導権の切り替えが行われる。

【 0 2 9 4 】

サーバは、Farewell Protocol が終了したと判断したとき、ステップ S 3 4 6 において、クライアントに対して SERVER_GOOD_BYE_DONE メッセージを送信し、コネクションを終了させる。これにより、クライアントのリソースが開放され、処理が終了される。

【 0 2 9 5 】

このように、SERVER_GOOD_BYE_DONE メッセージが送信されることにより、処理が終了される場合もある。

【 0 2 9 6 】

図 2 6 は、上述したように、クライアントからサーバに対して、CLIENT_HELLO_DONE メッセージが送信されることにより、コネクションが終了される場合のシーケンスである。

【 0 2 9 7 】

すなわち、ステップ S 3 6 1 において、クライアントからサーバに対して、CLIENT_HELLO メッセージが送信され、ステップ S 3 6 2 に進み、DEVICES メッセー

ジが送信される。

【 0 2 9 8 】

ステップ S 3 6 3 において、アプリケーション固有のメッセージが送信され、ステップ S 3 6 4 に進み、CLIENT_HELLO_DONEメッセージが送信されたとき、サーバとクライアント間でのコネクションが確立される。

【 0 2 9 9 】

なお、サーバにおいては、ステップ S 3 7 1 乃至 S 3 7 4 において、クライアントから送信されてきたCLIENT_HELLOメッセージ、DEVICESメッセージ、アプリケーション固有のメッセージ、および、CLIENT_HELLO_DONEメッセージがそれぞれ受信される。

【 0 3 0 0 】

例えば、クライアントが携帯電話機などの機器である場合、図 8 の処理と比較して、データの送受信が一部省略された、以上の処理により、クライアントとサーバの間でコネクションが確立されることがある。

【 0 3 0 1 】

図 2 7 は、上述したように、サーバからクライアントに対して、SERVER_GOOD_BYE_DONEメッセージが送信されることにより、Farewell Protocolが終了される場合のシーケンスである。

【 0 3 0 2 】

すなわち、ステップ S 3 9 1 において、サーバからSERVER_GOOD_BYEメッセージが送信されることによりFarewell Protocolが開始され、ステップ S 3 9 2 において、RETURN_CODEメッセージが送信される。また、ステップ S 3 9 3 において、アプリケーション固有のメッセージが送信される。

【 0 3 0 3 】

ステップ S 3 9 4 において、サーバからクライアントに対して、SERVER_GOOD_BYE_DONEメッセージが送信され、これにより、コネクションが終了される。

【 0 3 0 4 】

サーバから送信されたSERVER_GOOD_BYEメッセージ、RETURN_CODEメッセージ、アプリケーション固有のメッセージ、および、SERVER_GOOD_BYE_DONEメッセージ

は、それぞれ、クライアントにより、ステップ S 3 8 1 乃至 S 3 8 4 の処理により受信される。SERVER_GOOD_BYE_DONE メッセージが受信されたとき、クライアントのリソースが開放される。

【 0 3 0 5 】

上述した Handshake Protocol の場合と同様に、例えば、クライアントが携帯電話機などの機器である場合、図 1 0 の処理と比較して、データの送受信が一部省略された、以上のような処理により、クライアントとサーバの間でのコネクションが終了されることがある。

【 0 3 0 6 】

上述した一連の処理は、ハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。

【 0 3 0 7 】

一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば、汎用のパーソナルコンピュータなどに、ネットワークや記録媒体からインストールされる。

【 0 3 0 8 】

この記録媒体は、図 2 に示されるように、装置本体とは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク 4 1 (フレキシブルディスクを含む)、光ディスク 4 2 (CD-ROM (Compact Disk-Read Only Memory), DVD (Digital Versatile Disk) を含む)、光磁気ディスク 4 3 (MD (登録商標) (Mini-Disk) を含む)、もしくは半導体メモリ 4 4 などよりなるパッケージメディアにより構成されるだけでなく、装置本体に予め組み込まれた状態でユーザに提供される、プログラムが記録されている ROM 3 2 や、記憶部 3 8 に含まれるハードディスクなどで構成される。

【 0 3 0 9 】

なお、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に従って時系列的に行われる処理はもちろん、必ずしも時

系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【0310】

また、本明細書において、システムとは、複数の装置により構成される装置全体を表わすものである。

【0311】

【発明の効果】

本発明によれば、情報処理装置と情報処理端末の間で通信を行うこと、特に、重宝処理装置が主導となり情報処理端末との間で通信を行うことができる。

【0312】

また、本発明によれば、効率的にかつ確実に各種のパケットを送受信することができるとともに、通信コストを抑制することができ、これにより、情報処理装置が、情報処理端末を効率的かつ確実に制御することができる。

【0313】

さらに、本発明によれば、処理のエラーが発生した場合であっても、それに続く処理をクライアントに実行させることができる。

【図面の簡単な説明】

【図1】

本発明を適用した制御システムの構成例を示す図である。

【図2】

図1のサーバの構成例を示すブロック図である。

【図3】

サーバの機能構成例を示すブロック図である。

【図4】

クライアントの機能構成例を示すブロック図である。

【図5】

制御プロトコルの状態遷移について説明する図である。

【図6】

パケットのフォーマットの例を示す図である。

【図 7】

Handshake Protocolにおいて用いられるメッセージを示す図である。

【図 8】

図 7 に示されるメッセージを用いて行われるHandshake Protocolについて説明するシーケンスである。

【図 9】

Farewell Protocolにおいて用いられるメッセージを示す図である。

【図 1 0】

図 9 に示されるメッセージを用いて行われるFarewell Protocolについて説明するシーケンスである。

【図 1 1】

Error Protocolにおいて用いられるメッセージを示す図である。

【図 1 2】

図 1 1 に示されるメッセージを用いて行われるError Protocolについて説明するシーケンスである。

【図 1 3】

図 1 1 に示されるメッセージを用いて行われる他のError Protocolについて説明するシーケンスである。

【図 1 4】

Update Entity Protocolにおいて用いられるメッセージを示す図である。

【図 1 5】

図 1 4 に示されるメッセージを用いて行われるUpdate Entity Protocolについて説明するシーケンスである。

【図 1 6】

Application Data Transfer Protocolにおいて用いられるメッセージを示す図である。

【図 1 7】

図 1 6 に示されるメッセージを用いて行われるApplication Data Transfer Protocolについて説明するシーケンスである。

【図 1 8】

Operate Entity Protocolにおいて用いられるメッセージを示す図である。

【図 1 9】

図 1 8 に示されるメッセージを用いて行われる Operate Entity Protocol について説明するシーケンスである。

【図 2 0】

Neutral ステートにおける通信を説明するシーケンスである。

【図 2 1】

Neutral ステートにおける他の通信を説明するシーケンスである。

【図 2 2】

Neutral ステートにおけるさらに他の通信を説明するシーケンスである。

【図 2 3】

Exchange Protocol について説明するシーケンスである。

【図 2 4】

Exchange Protocol について説明するシーケンスである。

【図 2 5】

制御プロトコルの他の状態遷移について説明する図である。

【図 2 6】

Handshake Protocol について説明する他のシーケンスである。

【図 2 7】

Farewell Protocol について説明する他のシーケンスである。

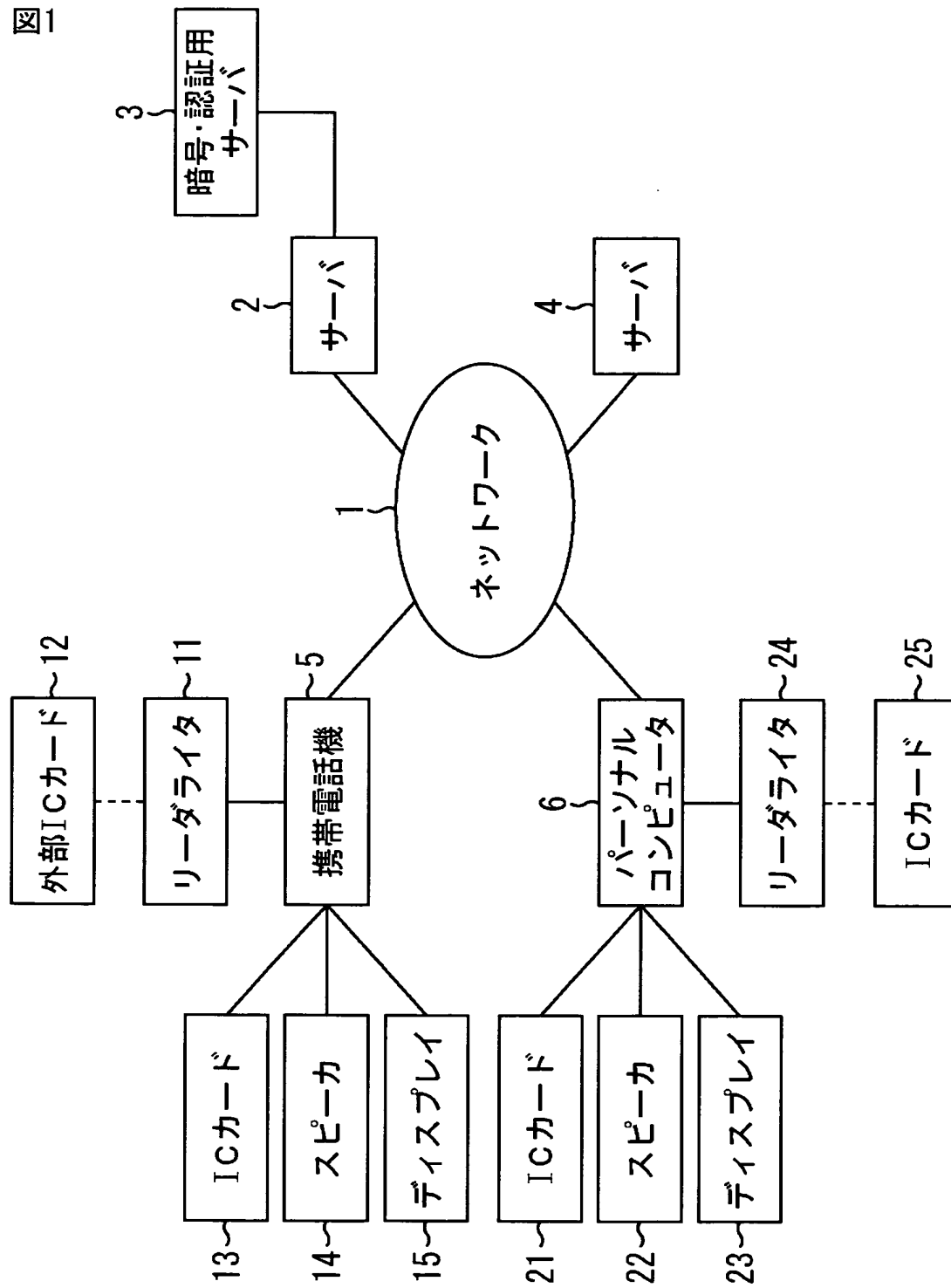
【符号の説明】

1 ネットワーク, 2 および 4 サーバ, 5 携帯電話機, 6 パーソナルコンピュータ, 11 磁気ディスク, 12 光ディスク, 13 光磁気ディスク, 14 半導体メモリ

【書類名】 図面

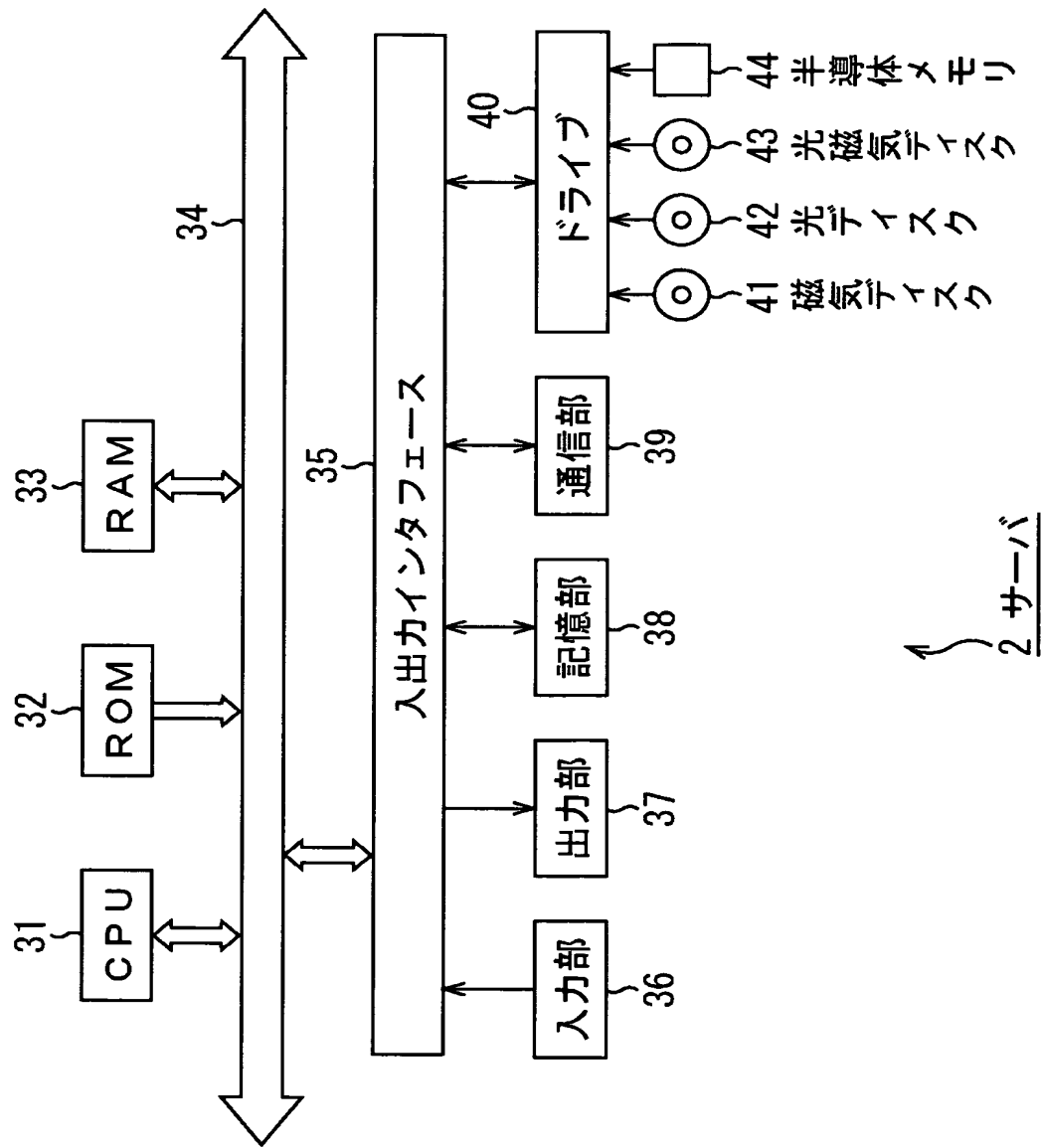
【図 1】

図1



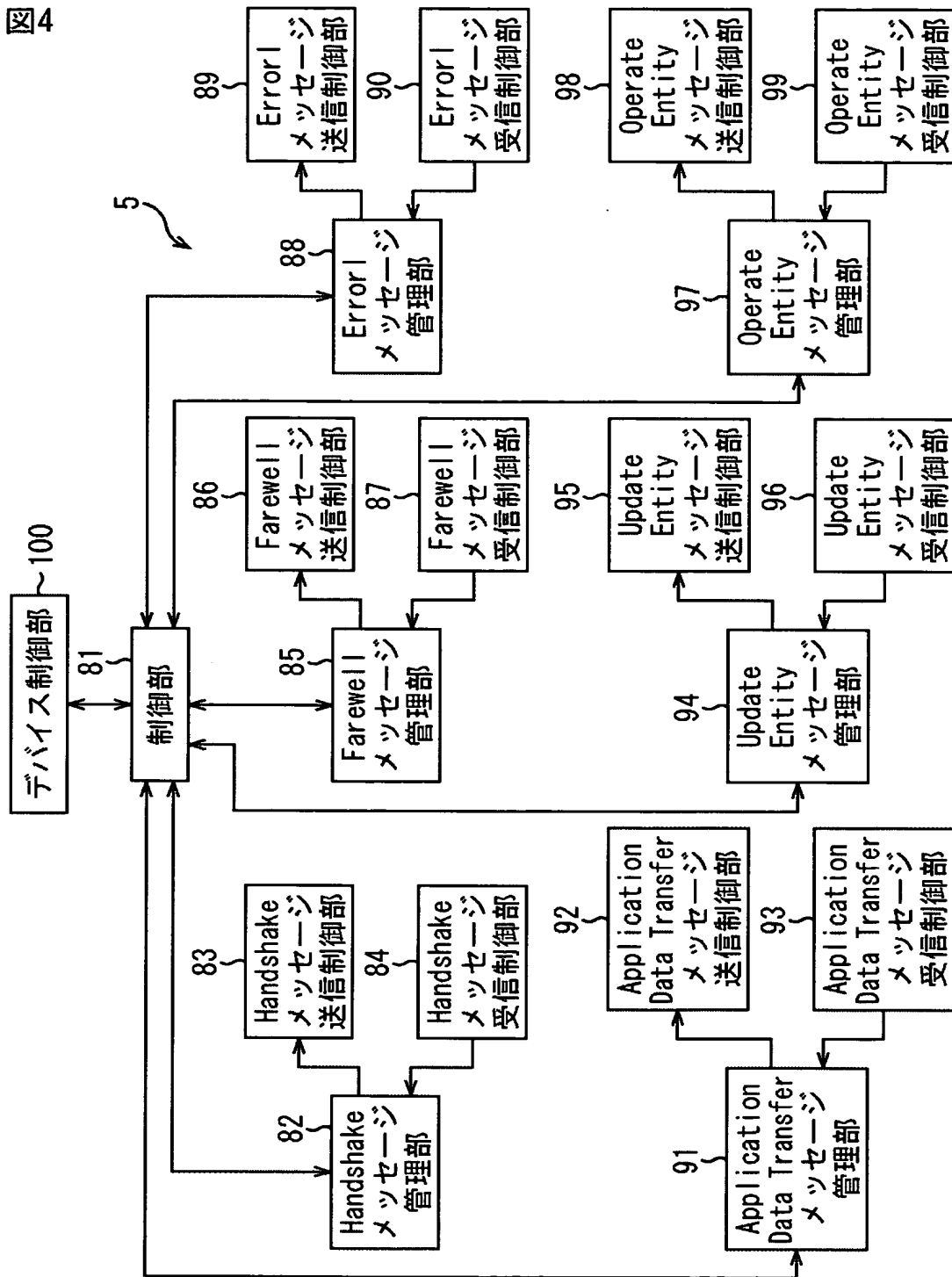
【図 2】

図2

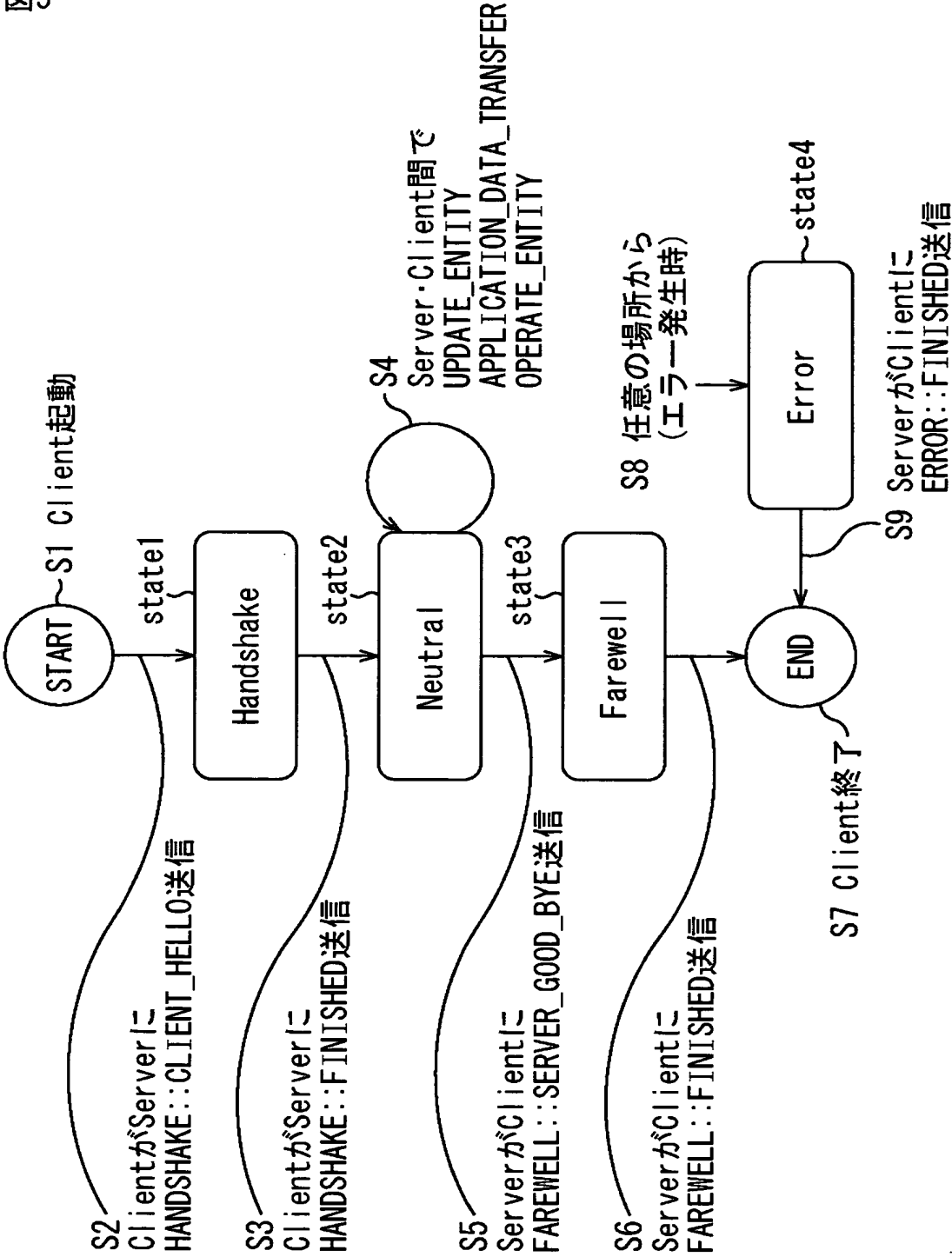


【図 4】

図4



【図5】
図5

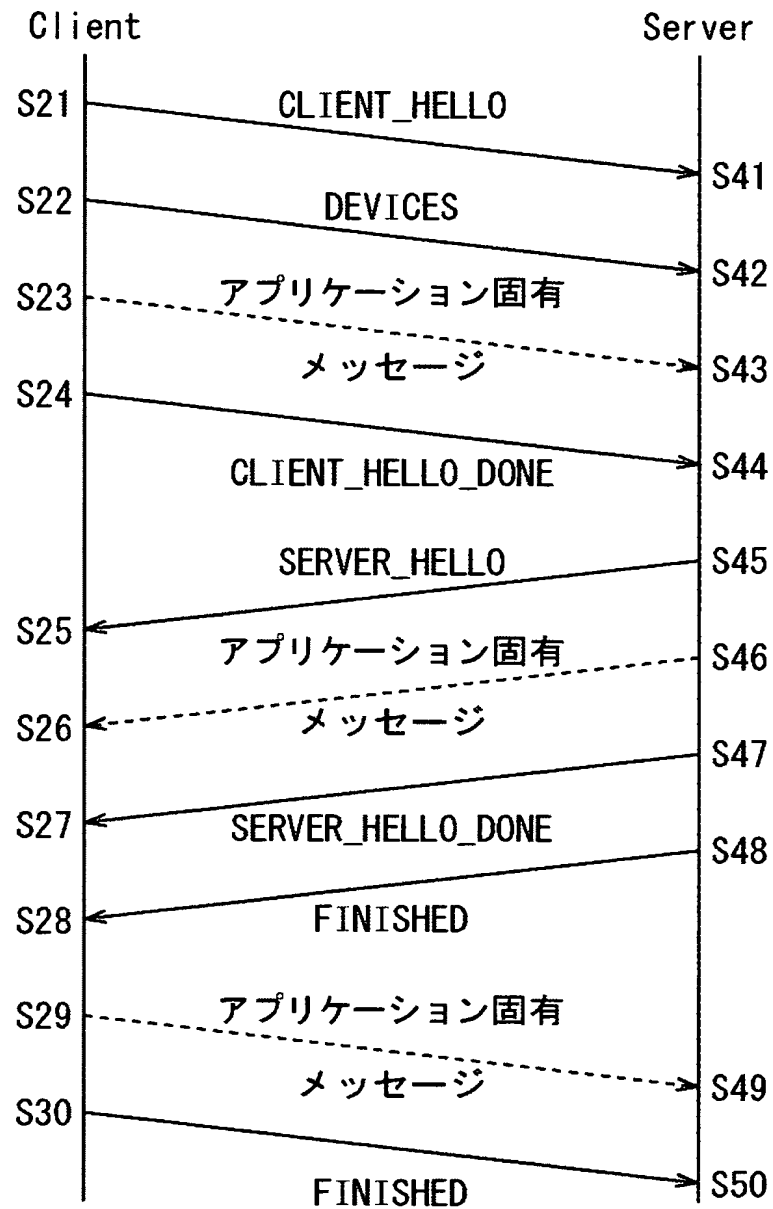


【図 7】
図7

Handshake Protocol		
メッセージ名	意味	データ部
FINISHED	サブプロトコル終了	なし
WARNING	警告通知	警告メッセージ: 文字列
CLIENT_HELLO	サーバ宛Handshake データ送信開始	なし
CLIENT_HELLO_DONE	サーバ宛Handshake データ送信終了	なし
SERVER_HELLO	クライアント宛Handshake データ送信開始	なし
SERVER_HELLO_DONE	クライアント宛Handshake データ送信終了	なし
DEVICES	デバイスリストの通知	・デバイスID ・パーミッション ・タイプ名長 ・タイプ名 ・デバイス名長 ・デバイス名

【図 8】

図8

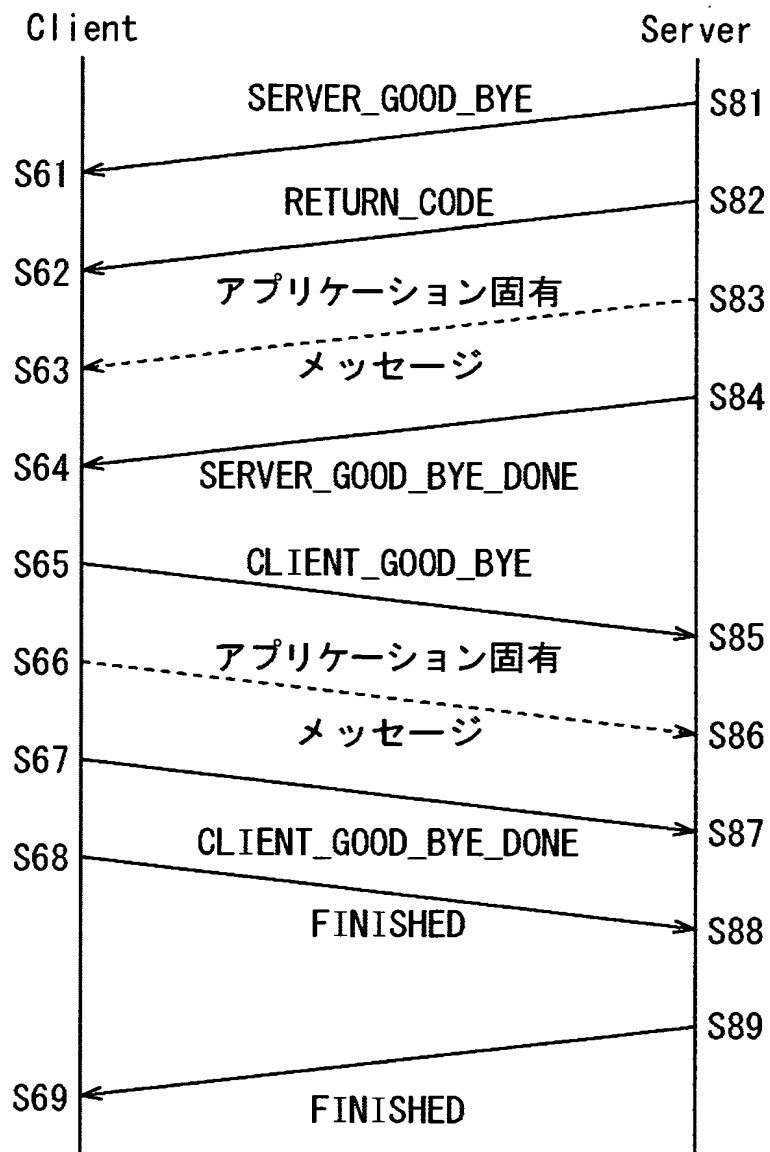


【図 9】
図9

Farewell Protocol		
メッセージ名	意味	データ部
FINISHED	サブプロトコル終了	なし
WARNING	警告通知	警告メッセージ:文字列
CLIENT_GOOD_BYE	サーバ宛Farewell データ送信開始	なし
CLIENT_GOOD_BYE_DONE	サーバ宛Farewell データ送信終了	なし
SERVER_GOOD_BYE	クライアント宛Farewell データ送信開始	なし
SERVER_GOOD_BYE_DONE	クライアント宛Farewell データ送信終了	なし
RETURN_CODE	サーバ終了コード通知	終了コード

【図10】

図10

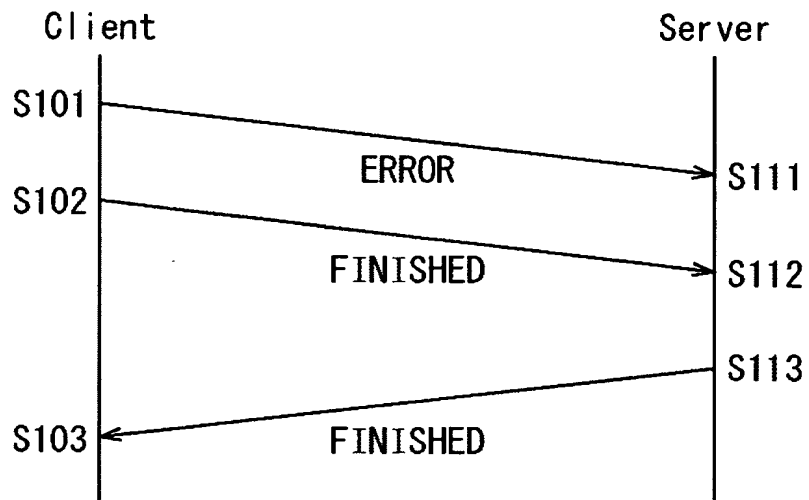


【図 1 1】
図11

Error Protocol		
メッセージ名	意味	データ部
FINISHED	サブプロトコル終了	なし
WARNING	警告通知	警告メッセージ: 文字列
PACKET_FORMAT_ERROR	フォーマットエラー	エラーメッセージ: 文字列
ILLEGAL_STATE_ERROR	シーケンス的に不正なメッセージを検知	エラーメッセージ: 文字列
UNEXPECTED_ERROR	上記以外の予期しないエラーが発生した	エラーメッセージ: 文字列

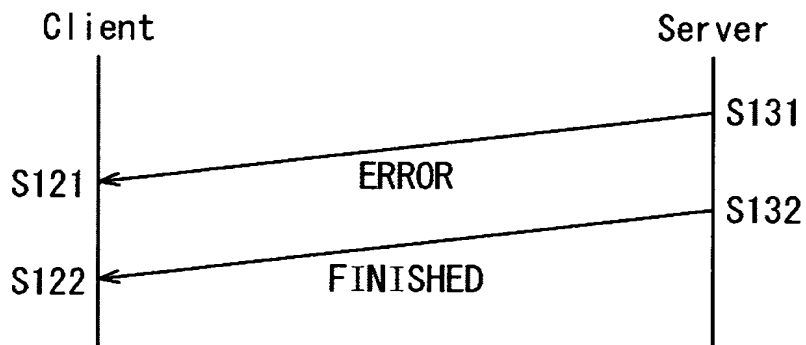
【図 12】

図12



【図 13】

図13



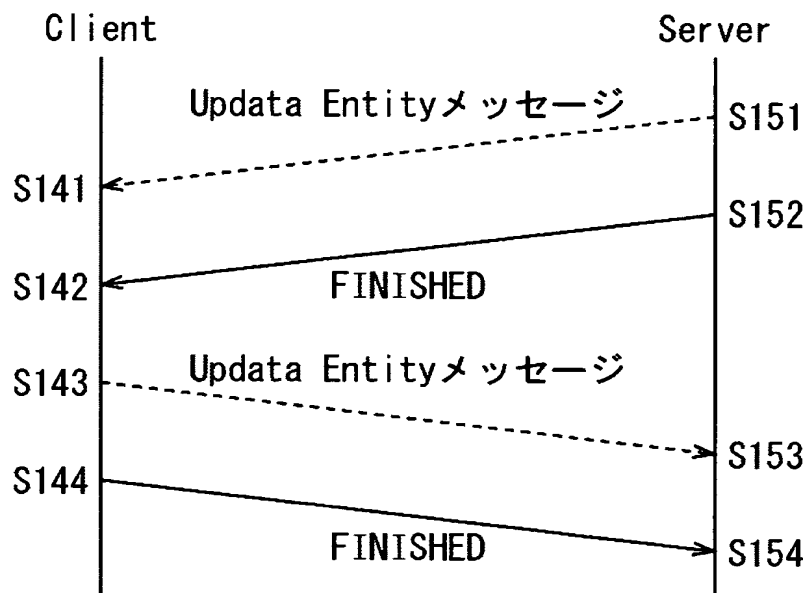
【図 1 4】

図 14

Update Entity Protocol		
メッセージ名	意味	データ部
FINISHED	パケットの終端	なし
WARNING	警告通知	警告メッセージ: 文字列
SET_PROPERTY	クライアント側デバイスの属性値を設定する 返信: PROPERTY_SET 失敗時: 処理中止	・属性名長 ・属性名 ・属性値長 ・属性値
PROPERTY_SET	SET_PROPERTY に対する返信	成功フラグ
GET_PROPERTY	クライアント側デバイスの属性値を取得する 返信: PROPERTY 失敗時: 処理中止	・属性名長 ・属性名
PROPERTY	GET_PROPERTY に対する返信	・属性値長 ・属性値
SET_NETWORK_TIMEOUT	クライアントのネットワークタイムアウト 値を設定する。 返信: なし 失敗時: 処理続行	・タイムアウト値 (msec)

【図15】

図15



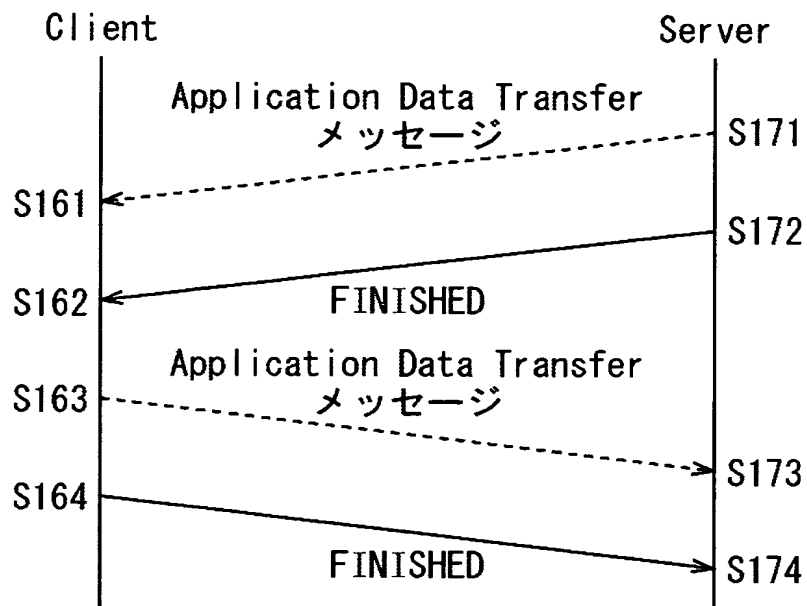
【図16】

図16

Application Data Transfer Protocol	
メッセージ名	意味
FINISHED	パケットの終端
WARNING	警告通知

【図 17】

図17



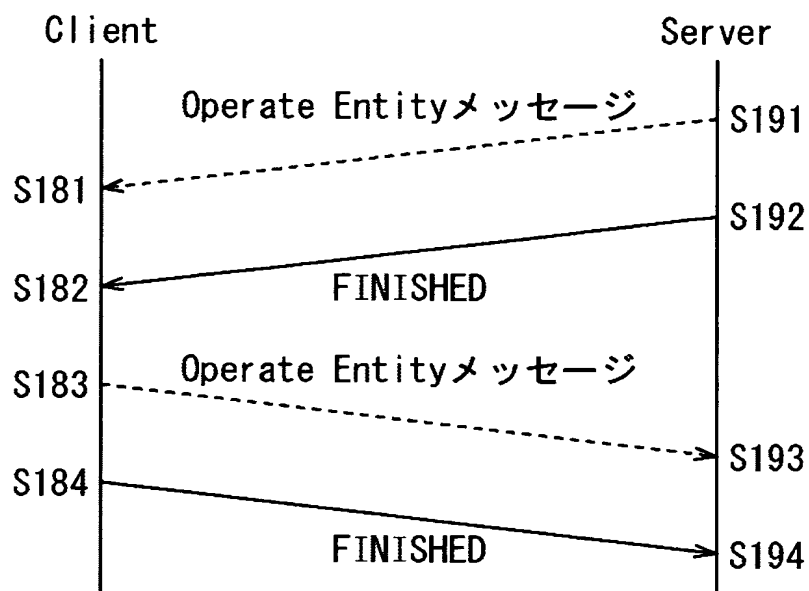
【図 18】

図 18

Operate Entity Protocol		
メッセージ名	意味	データ部
FINISHED	パケットの終端	なし
WARNING	警告通知	警告メッセージ: 文字列
UPDATE_VIEW	ビューの更新 返信: VIEW_UPDATED 失敗時: 処理中止	なし
VIEW_UPDATED	UPDATE_VIEW に対する返信	成功フラグ
BEGIN_DATA_INPUT	データ入力開始 返信: END_DATA_INPUT 失敗時: 処理中止	なし
END_DATA_INPUT	BEGIN_DATA_INPUT に対する返信	<ul style="list-style-type: none"> ・成功フラグ ・属性名長 ・属性名 ・属性値長 ・属性値
OPERATE_DEVICE	クライアント側のデバイスを操作する 返信: DEVICE_RESPONSE 失敗時: 処理中止	<ul style="list-style-type: none"> ・操作名文字数 ・操作名 ・パラメータ長 ・パラメータ
DEVICE_RESPONSE	デバイス操作結果	<ul style="list-style-type: none"> ・レスポンス長 ・レスポンス
PLAY_SOUND	クライアント側で指定した論理名に対応した音を鳴らす。 返信: なし 失敗時: 処理続行	音の論理名: 文字列

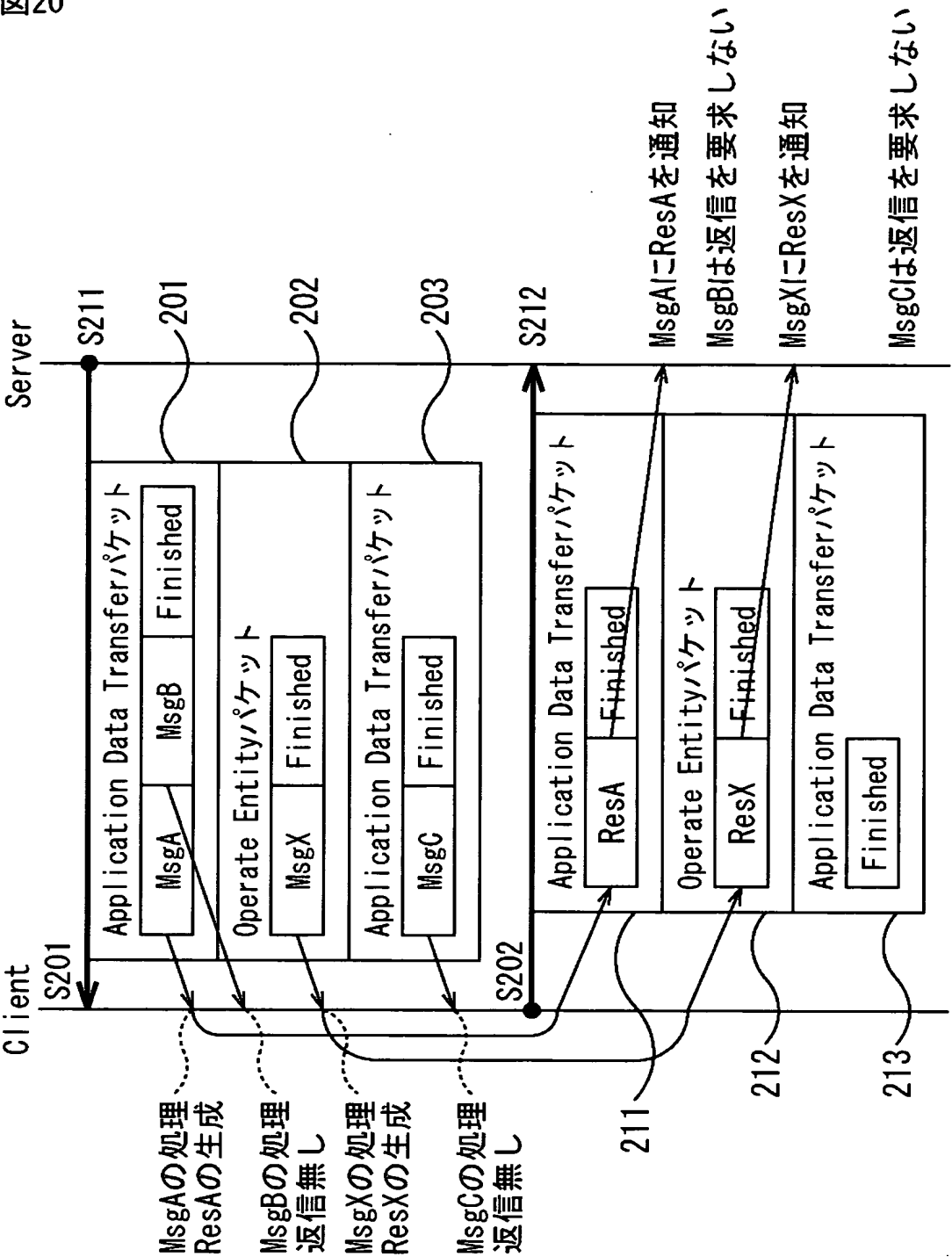
【図 19】

図19



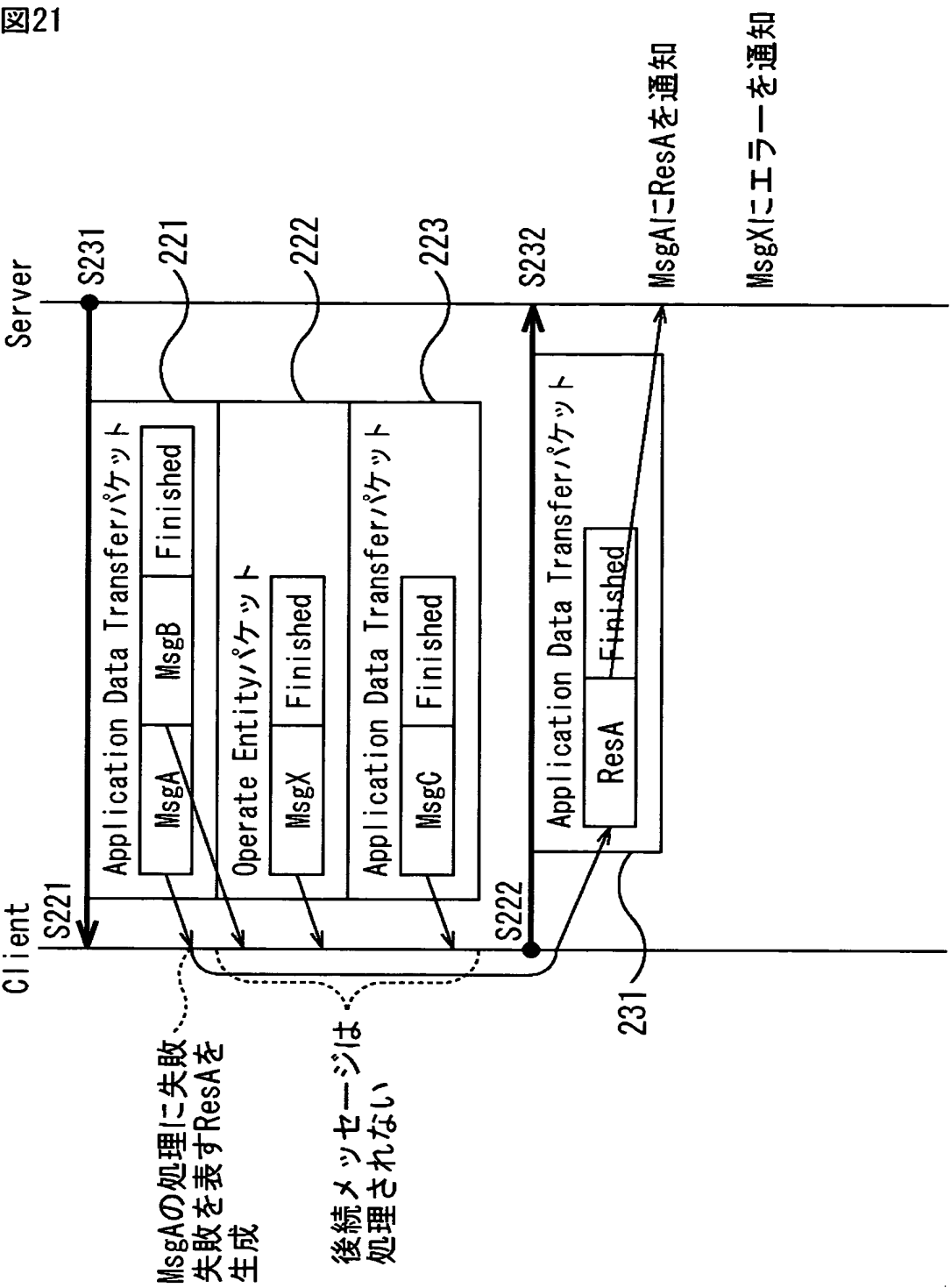
【図 20】

図20



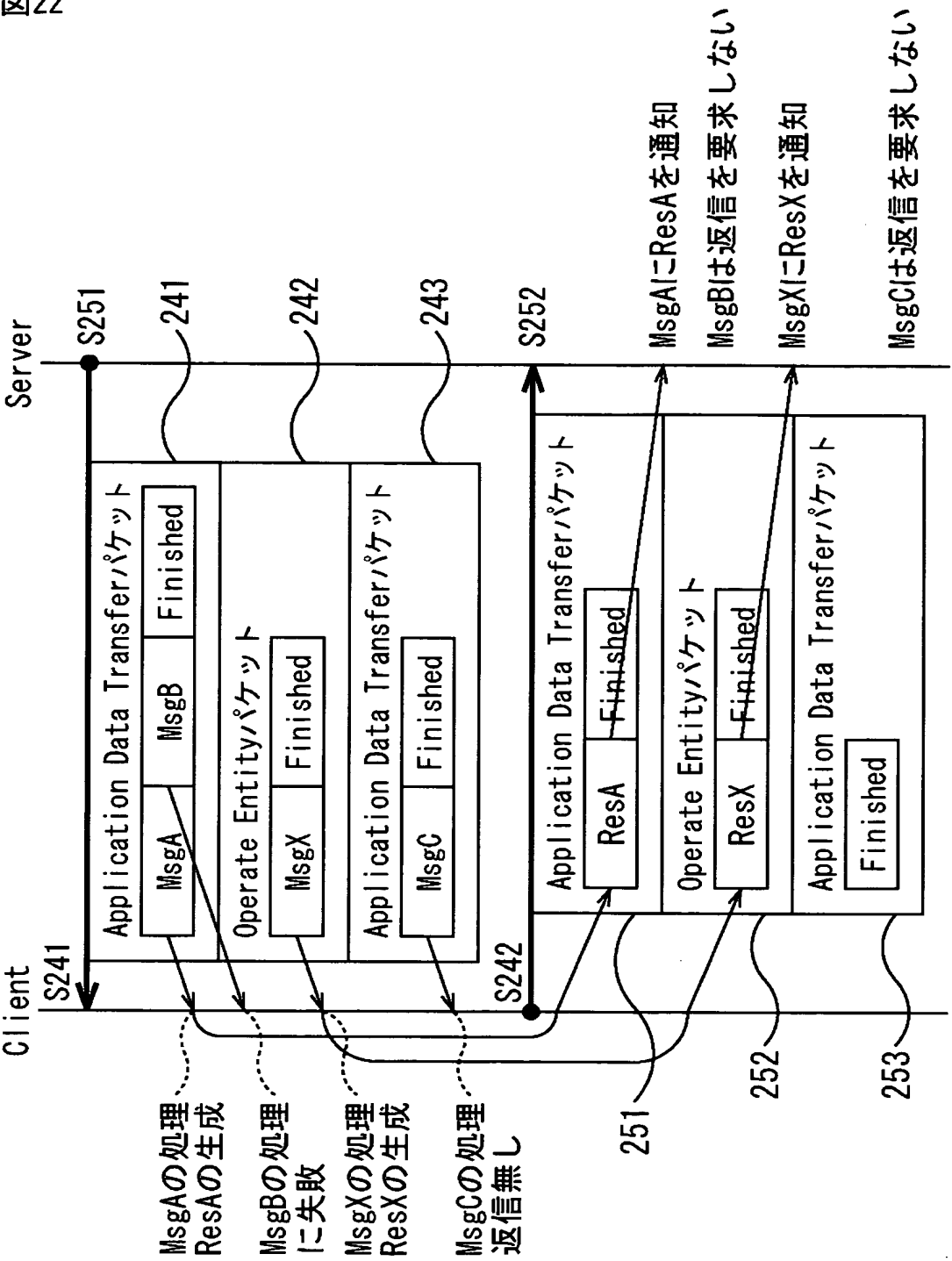
【図 21】

図21



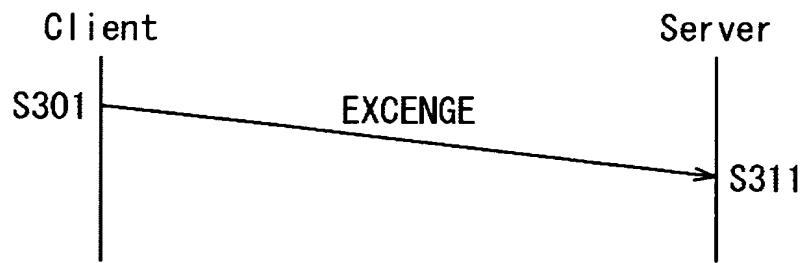
【図 22】

図22



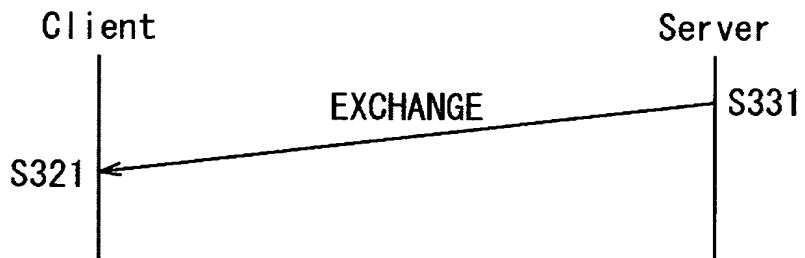
【図 2 3】

図23



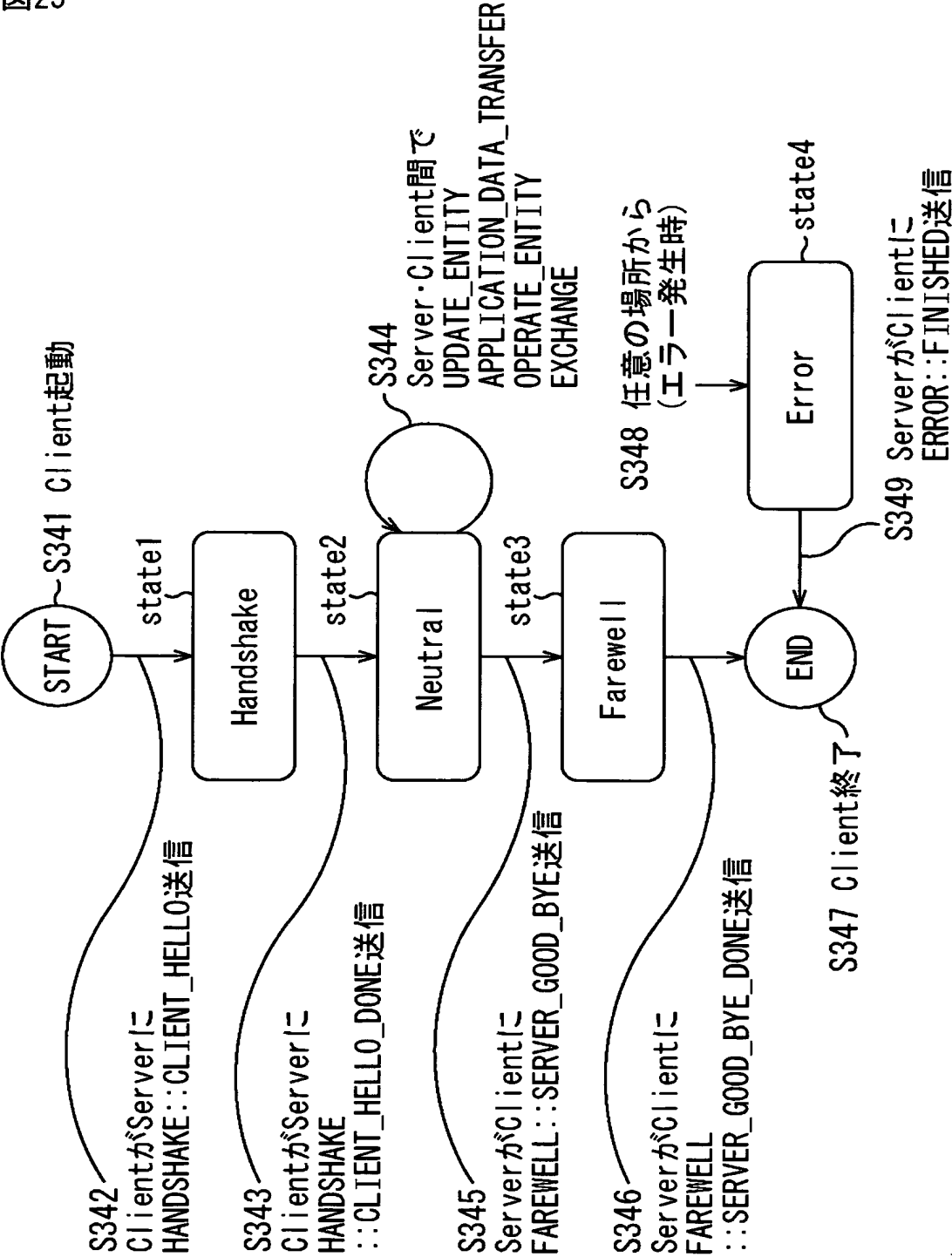
【図 2 4】

図24



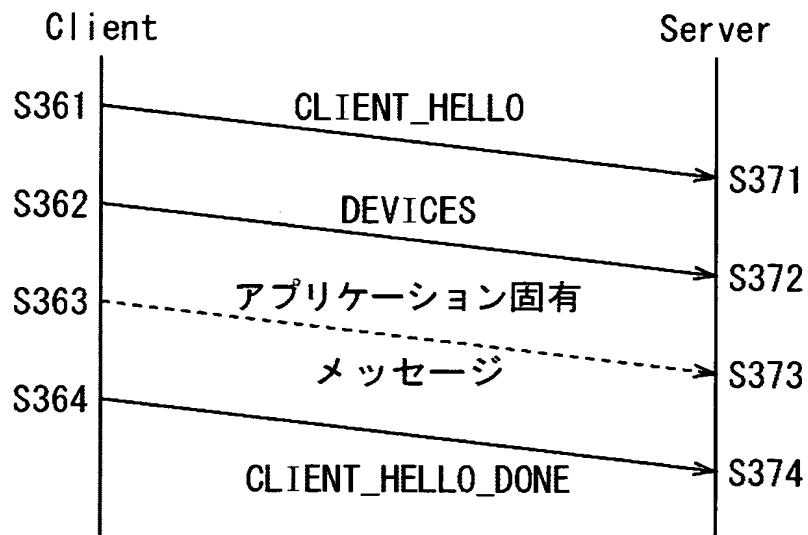
【図 25】

図25



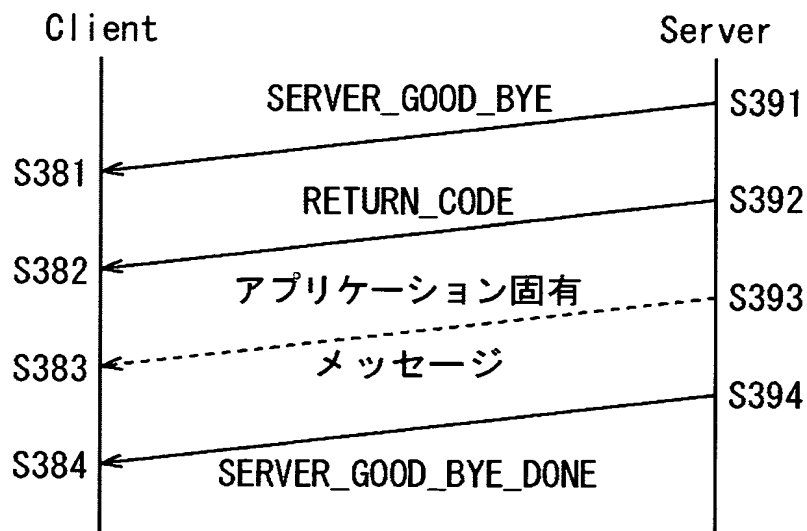
【図 26】

図26



【図 27】

図27



【書類名】 要約書

【要約】

【課題】 ネットワークを介して接続される機器の制御を、より確実に、かつ効率的に行うことができるようにする。

【解決手段】 ICカード3などのデバイスが接続された携帯電話機5（クライアント）によりHandshake Protocolが開始され、サーバ2に対して通信の開始が要求される。Handshake Protocolが行われ、通信が確立したとき、通信の主導権がサーバ側に移動した状態（Neutralステート）になる。Neutralステートにおいては、サーバ2からクライアントに対して、所定の数のメッセージと1つのFINISHEDメッセージを含む制御パケットが送信される。制御パケットを受信したクライアントにおいては、メッセージに基づいてデバイスに対する処理が行われる。本発明は、デバイスを制御するクライアント装置（PC、携帯電話機、PDA、家庭電化製品等）と、サーバ装置の、システムの双方の機器に適用可能である。

【選択図】 図1

特願 2 0 0 2 - 3 4 9 9 9 4

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 2 1 8 5]

1. 変更年月日

1 9 9 0 年 8 月 3 0 日

[変更理由]

新規登録

住 所

東京都品川区北品川 6 丁目 7 番 3 5 号

氏 名

ソニー株式会社